



PUBLIC

2024-01-04

# SAP Cloud Identity Services - Identity Provisioning

# Content

<b>1</b>	<b>SAP Cloud Identity Services – Identity Provisioning</b>	<b>5</b>
1.1	What Is Identity Provisioning?	5
	Tenant Model	8
	Tenant Infrastructure	10
	Regional Availability	11
	Disaster Recovery/High Availability	15
	Accessibility Features in Identity Provisioning	17
1.2	What's New for Identity Provisioning	17
	Release Notes – 2021	63
	Release Notes – 2020	63
	Release Notes – 2019	63
	Release Notes – 2018	63
	Release Notes – 2017	68
	Release Notes – 2016	82
1.3	Concepts	85
	System Types	86
	Properties	90
	Transformations	323
1.4	Initial Setup of Bundle Tenants	406
	Obtain a Bundle Tenant	407
	Getting a Trial Tenant	411
	Access Identity Provisioning UI of Bundle Tenants	413
	Provisioning Systems for Bundle Tenants	416
1.5	Bundle Tenants and Connectors	422
	SAP Business Technology Platform Bundle	427
	SAP Commissions Bundle	430
	SAP Cloud Identity Access Governance Bundle	431
	SAP Integrated Business Planning for Supply Chain Bundle	432
	SAP Jam Collaboration Bundle	435
	SAP Marketing Cloud Bundle	437
	SAP SuccessFactors Bundle	438
	SAP SuccessFactors Learning Bundle	440
	SAP S/4HANA Cloud Bundle	442
	SAP Build Work Zone, advanced edition Bundle	444
	Sales Cloud – Analytics & AI Bundle	446
	SAP Concur Bundle	447

	SAP Fieldglass Bundle. . . . .	448
	SAP Business Ecology Management Bundle. . . . .	449
	SAP Commerce Cloud Bundle. . . . .	450
1.6	Supported Systems. . . . .	452
	Source Systems. . . . .	452
	Target Systems. . . . .	702
	Proxy Systems. . . . .	981
1.7	Operations. . . . .	1476
	Add a System. . . . .	1477
	Search and Edit a System. . . . .	1480
	Delete a System. . . . .	1481
	Enable and Disable Systems. . . . .	1482
	Export and Import Systems. . . . .	1482
	Update Connector Version. . . . .	1484
	Manage Authorizations. . . . .	1487
	Manage Transformations. . . . .	1494
	Manage Properties . . . . .	1505
	Manage Certificates. . . . .	1506
	Connecting to On-Premise Systems. . . . .	1512
	Manage Full and Delta Read. . . . .	1519
	Manage Deleted Entities. . . . .	1522
	Start and Stop Provisioning Jobs. . . . .	1524
	Handle Rate Limits. . . . .	1532
	Handle Failed Operations. . . . .	1533
	Configure Identity Provisioning in SAP Cloud Identity Services Administration Console. . . . .	1535
	Migrate Identity Provisioning Bundle Tenant. . . . .	1536
	Reset Identity Provisioning Tenant. . . . .	1542
	Reset Identity Provisioning System. . . . .	1542
1.8	Security. . . . .	1544
	Communication Security. . . . .	1545
	Customer Data. . . . .	1545
	Authentication and Roles. . . . .	1547
	Job Logs. . . . .	1548
	Data Protection and Privacy. . . . .	1548
1.9	Specific Scenarios. . . . .	1554
	Real-Time Provisioning. . . . .	1554
	Hybrid Scenario: SAP Identity Management. . . . .	1565
	Identity Directory. . . . .	1567
1.10	Monitoring and Troubleshooting. . . . .	1593
	Monitor Provisioning Job Logs. . . . .	1594
	Monitor Real-Time Logs. . . . .	1598

	Manage Provisioning Job Logs. . . . .	1600
	Manage Job Notifications. . . . .	1605
	Access Audit Logs. . . . .	1607
1.11	Standalone Tenants. . . . .	1608
	Use a Standalone Tenant. . . . .	1612
	Access Identity Provisioning UI of Standalone Tenants. . . . .	1614
1.12	Service Offboarding. . . . .	1617
	Reset the Identity Provisioning (Bundles). . . . .	1618
	Reset/Remove the Identity Provisioning (Standalone). . . . .	1618
1.13	Submitting Improvement Requests. . . . .	1619
1.14	Getting Support. . . . .	1620

# 1 SAP Cloud Identity Services – Identity Provisioning



## Get Started



## What's New

[What Is Identity Provisioning? \[page 5\]](#)

[Tenant Model \[page 8\]](#)

[Initial Setup of Bundle Tenants \[page 406\]](#)

[Release Notes – 2021 \[page 17\]](#)

[Release Notes – 2020](#)

[Release Notes – 2019](#)

[Release Notes – 2018 \[page 63\]](#)

[Release Notes – 2017 \[page 68\]](#)

[Release Notes – 2016 \[page 82\]](#)



## Scenarios



## Resources

[Supported Systems \[page 452\]](#)

[Local Identity Directory \[page 1568\]](#)

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[Real-Time Provisioning \[page 1554\]](#)

[Operations \[page 1476\]](#)

[Security \[page 1544\]](#)

[Monitoring and Troubleshooting \[page 1593\]](#)

[Getting Support \[page 1620\]](#)

[Disclaimer](#)

[Legal Disclosure](#)

[Copyright and Trademarks](#)

## 1.1 What Is Identity Provisioning?

Manage identity lifecycle processes for cloud and on-premise systems.

The Identity Provisioning service automates identity lifecycle processes. It helps you provision identities and their authorizations to various cloud and on-premise business applications.

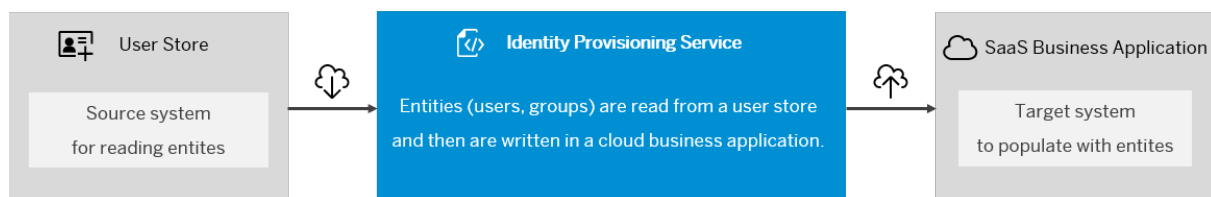
## Environment

Identity Provisioning tenants run on the infrastructure of SAP Cloud Identity Services and the SAP BTP, Neo environment.

## Features

<b>User and Group Provisioning</b>	Provision users and groups between multiple supported cloud and on-premise systems, both SAP and non-SAP.
<b>User and Group Filtering</b>	Configure default transformations or filtering properties to control what data to be provisioned and what to be skipped.
<b>Full and Delta Read Mode</b>	Run a provisioning job in full mode to read all entities from a source system, or in delta read mode - to read only the modified data.
<b>Job Logging</b>	View and export job logs from the Identity Provisioning administration console. Logs display details about the job status and the provisioned entities.
<b>Notifications</b>	Subscribe to a source system to receive notifications for the status of provisioning jobs.

## Overview Graphic



## Use Cases

Identity Provisioning supports the following use cases:

- [Provisioning from Source to Target Systems \[page 86\]](#)  
The main use case of Identity Provisioning is to read users and groups from a source system and provision them to a target system. Filtering and/or mapping are applied during job execution.
- [Hybrid Integration with Identity Management Systems \[page 1565\]](#)  
Identity Provisioning can be used for integrating cloud solutions with on-premise or cloud identity management systems that support SCIM 2.0 standard, such as SAP Identity Management and SAP Cloud Identity Access Governance.

In a hybrid integration scenario, Identity Provisioning acts as a proxy between a cloud solution and an on-premise or cloud system. This means the Identity Provisioning is used for configuring and exposing the cloud solution as a proxy system and connect it to the external identity management system without making a direct connection between them.

- [Real-Time Provisioning from Identity Authentication \[page 1554\]](#)

Identity Provisioning can be used for immediate, real-time provisioning of Identity Authentication users to any target system. Unlike the standard provisioning, where reading and writing of users is triggered by jobs, real-time provisioning is triggered by events (such as, user self-registration or user modification in Identity Authentication).

- [Storing Users and Groups in Local Identity Directory \[page 1567\]](#)

Identity Provisioning is mainly used for provisioning users and groups. However, it can also be used for storing users and groups when a specific type of system - Local Identity Directory, is configured. In a typical use case, the Local Identity Directory is first configured as a target system, where users and groups are provisioned to, and then configured as a source system, from where users and groups are read and provisioned to target systems.

The identity directory provides a System for Cross-domain Identity Management (SCIM) 2.0 REST API for managing resources (users, groups, and custom schemas).

#### **i Note**

The *Local Identity Directory* connector is available for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

## Prerequisites

To use Identity Provisioning, you need to obtain a tenant. The service provides two types of tenants - bundle and standalone.

For more information, see:

- [Obtain a Bundle Tenant \[page 407\]](#)
- [Use a Standalone Tenant \[page 1612\]](#)

## Tools

You can access Identity Provisioning administration console as an HTML5 application. Depending on your Identity Provisioning tenant type, you can do this as follows:

- [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
- [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

#### **⚠ Caution**

**Effective October 20, 2020, Identity Provisioning is offered bundled with SAP cloud solutions.** You can obtain and use it, along with Identity Authentication, as part of a bundled SAP cloud solution that you need to purchase. The service is no longer sold as a standalone product. Existing customers of standalone Identity Provisioning can use it as-is until the end of their contracts.

To check the list of SAP cloud solutions that bundle Identity Provisioning, see [Bundle Tenants and Connectors \[page 422\]](#)

## Regional Availability

You can access Identity Provisioning tenants on the infrastructure of SAP Cloud Identity Services and the SAP BTP, Neo environment.

For more information, see: [Regional Availability \[page 11\]](#).

### 1.1.1 Tenant Model

SAP Cloud Identity Services – Identity Provisioning provides two types of tenants - **bundle** and **standalone**.

Although bundle and standalone tenants differ in various aspects: pricing (in bundle tenants, Identity Provisioning is free of charge), connectors availability and level of access to SAP BTP cockpit, the provisioning functionality remains the same.

Both type of tenants can run on SAP Cloud Identity Services infrastructure and SAP BTP, Neo Environment.

## Bundle Tenant

A bundle tenant is an instance of Identity Provisioning that comes with a set of preconfigured provisioning systems relevant to one or more bundled SAP cloud solutions.

### ⚠ Caution

**Effective March 15, 2022, new Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only.** Existing customers of bundle tenants on Neo environment can continue using them as-is. For more information, see: [Tenant Infrastructure \[page 10\]](#)

When an SAP cloud solution bundles with SAP Cloud Identity Services, you are entitled to receive Identity Authentication and Identity Provisioning tenants without additional costs on the purchase of the corresponding SAP cloud solution's license. These Identity Authentication and Identity Provisioning tenants come preconfigured with the SAP cloud solution. For more information, see [Obtain a Bundle Tenant \[page 407\]](#)

You obtain Identity Provisioning bundle tenant with a set of provisioning systems (source, target and proxy) for which you have a license. Those systems are preconfigured in your tenant. Further usage of Identity Provisioning connectors and their availability depend on the infrastructure/environment your bundle tenant is running on. For more information, see [Bundle Tenants and Connectors \[page 422\]](#)

Regardless of how many SAP cloud solutions you have purchased, you are entitled to two Identity Provisioning bundle tenants – one for testing and one for productive purposes. For more information, see *Can I obtain additional bundle tenants?* in [Obtain a Bundle Tenant \[page 407\]](#)



Depending on the infrastructure or the environment your bundle tenant runs on, you can access and operate it as follows:

### SAP Cloud Identity Services Infrastructure

Bundle tenants created after March 15, 2022 run on SAP Cloud Identity Services infrastructure.

The Identity Provisioning admin access is fully controlled and configured in the administration console of Identity Authentication. This access is based on roles which are assigned to admin users in the [Users & Authorizations](#) screen of the Identity Authentication administration console. For more information, see [Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#)

### SAP BTP, Neo Environment

Bundle tenants created before March 15, 2022 run on SAP BTP, Neo environment.

Administrators of bundle tenants can only access their Identity Provisioning subaccount in SAP BTP cockpit to register OAuth clients, create connectivity destinations and configure Cloud Connector connections. This access is based on roles which are assigned to admin users in the [Authorization](#) tile of the Identity Provisioning administration console. For more information, see [Manage Authorizations in Neo Environment \[page 1490\]](#)

## Standalone Tenant

A standalone tenant allows you to use Identity Provisioning as a separate (standalone) product. For more information, see [Use a Standalone Tenant \[page 1612\]](#).

#### Caution

**Effective October 20, 2020, Identity Provisioning is offered bundled with SAP cloud solutions.** You can obtain and use it, along with Identity Authentication, as part of a bundled SAP cloud solution that you need to purchase. The service is no longer sold as a standalone product. Existing customers of standalone Identity Provisioning can use it as-is until the end of their contracts.

To check the list of SAP cloud solutions that bundle Identity Provisioning, see [Bundle Tenants and Connectors \[page 422\]](#)

The scope of the standalone tenant is not restricted. It can be used for provisioning of users and groups to and from all supported systems by Identity Provisioning service.

Depending on the infrastructure or the environment your standalone tenant runs on, you can access and operate it as follows:

### SAP Cloud Identity Services Infrastructure

Identity Provisioning service purchased between September 1, 2020 and October 20, 2020 runs on the infrastructure of SAP Cloud Identity Services.

You use a tenant that provides you access to both Identity Provisioning and Identity Authentication. You can access Identity Provisioning in all regions and data centers where the Identity Authentication is running. For more information, see [Access Identity Provisioning UI of Standalone Tenants -> SAP Cloud Identity Infrastructure \[page 1614\]](#)

## SAP BTP, Neo Environment

Identity Provisioning service purchased before September 1, 2020 runs on SAP BTP, Neo environment.

You access Identity Provisioning admin console by using SAP Business Technology Platform subaccounts via SAP BTP cockpit. You can access Identity Provisioning in all regions available for SAP BTP, Neo environment. For more information, see [Access Identity Provisioning UI of Standalone Tenants -> SAP BTP, Neo Environment \[page 1614\]](#)

## Related Information

[Identity Authentication: Tenant Model](#)

### 1.1.2 Tenant Infrastructure

Identity Provisioning bundle tenants can run on the infrastructure of SAP Cloud Identity Services and the SAP BTP, Neo environment.

#### ⚠ Caution

**Effective March 15, 2022, new Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only.** Existing customers of bundle tenants on SAP BTP, Neo environment can continue using them as-is.

Delivering bundle tenants on the infrastructure of SAP Cloud Identity Services improves the integration between the group of services that provide cloud identity capabilities: Identity Authentication, Identity Provisioning, and Identity Directory. The Identity Provisioning admin access is fully controlled and configured in the [Users & Authorizations](#) section of SAP Cloud Identity Services administration console, where customers can easily benefit from its numerous features, such as setting up single sign-on for corporate identity providers, enabling two-factor authentication and others.

Sharing the same infrastructure paves the way for tighter integration and common features in the future.

#### i Note

As of June 27, 2022, when the tenant migration from SAP BTP, Neo environment to SAP Cloud Identity infrastructure was released, all new Identity Provisioning features are delivered only for tenants running on SAP Cloud Identity infrastructure. For more information, see [Migrate Identity Provisioning Bundle Tenant \[page 1536\]](#).

## SAP Cloud Identity Infrastructure

Bundle tenants on this infrastructure come with the following specifics:

- The Identity Provisioning tenant URL uses the host of the corresponding Identity Authentication tenant of the customer. It follows the pattern: `https://<ias-host>/admin`.

For example: `https://best-run.accounts.ondemand.com/admin`

- The Identity Provisioning administrator authenticates to the corresponding Identity Authentication tenant of the customer with the admin user that has the [Manage Identity Provisioning](#) role enabled in SAP Cloud Identity Services administration console.  
Further Identity Provisioning administration access, such as authorizations to access API for real-time provisioning and access API for provisioning identities via proxy systems, is granted in the SAP Cloud Identity Services administration console. For more information, see [Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#)
- Almost all of the provisioning systems (connectors) supported by Identity Provisioning are enabled by default for bundle tenants on the infrastructure of SAP Cloud Identity Services. This means that, in addition to the automatically preconfigured systems relevant for a bundled SAP cloud solution, customers can manually configure the supported connectors as source, target and proxy systems in SAP Cloud Identity Services administration console. For more information, see [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#)

## SAP BTP, Neo Environment

Bundle tenants on this environment come with the following specifics:

- The Identity Provisioning tenant URL uses the bundle tenant ID and the region and host available for SAP BTP, Neo environment. It follows the pattern:  
`https://ips-<consumer_account>.dispatcher.<region_host>/webapp/index.html`, where `<consumer_account>` is the Identity Provisioning bundle tenantID.  
For example: `https://ips-a12345sdf678.dispatcher.cal.hana.ondemand.com/webapp/index.html`
- The Identity Provisioning administrator authenticates to the admin console of the service with his or her S-user credentials provided in the welcoming onboarding email from SAP. The admin user has the [Manage Identity Provisioning](#) role enabled in the Identity Provisioning admin console.  
Further Identity Provisioning administration access, such as authorizations to register OAuth clients, create connectivity destinations and configure Cloud Connector connections, is granted on the [Authorizations](#) screen in Identity Provisioning admin console. For more information, see [Manage Authorizations in Neo Environment \[page 1490\]](#)
- The set of provisioning systems enabled in bundle tenants on SAP BTP, Neo environment is restricted. The only exception is SAP Cloud Identity Access Governance bundle, which includes all supported provisioning systems by Identity Provisioning, except for Local Identity Directory.

### 1.1.3 Regional Availability

You can access Identity Provisioning in all regions available for SAP BTP, Neo environment. The only exception is - standalone tenants purchased between September 1, 2020 and October 20, 2020, which you can access in all regions and data centers where the Identity Authentication is running.

Tenant Type	Infrastructure of SAP Cloud		
	SAP BTP Neo Environment	Identity	Details
<b>Bundle tenants</b> (Created before 15.03.2022)	Yes	No	<p><b>Access:</b> All Neo regions and data centers. See:</p> <ul style="list-style-type: none"> <li>• <a href="#">Regions and Hosts (Neo)</a></li> <li>• <a href="#">SAP BTP Discovery Center: Identity Provisioning</a></li> </ul>
<b>Bundle tenants</b> (Created after 15.03.2022)	No	Yes	<p>Identity Provisioning and Identity Authentication are running on the same SAP Cloud Identity infrastructure.</p> <p><b>Access:</b> All regions and data centers where the Identity Authentication is running. See:</p> <ul style="list-style-type: none"> <li>• <a href="#">Identity Authentication: Regional Availability</a></li> <li>• <a href="#">SAP BTP Discovery Center: Identity Authentication</a></li> </ul>
<b>Standalone tenants</b> (Purchased before 01.09.2020)	Yes	No	<p>Existing customers can use standalone tenants as-is until the end of their contracts.</p> <p><b>Access:</b> All Neo regions and data centers. See:</p> <ul style="list-style-type: none"> <li>• <a href="#">Regions and Hosts (Neo)</a></li> <li>• <a href="#">SAP BTP Discovery Center: Identity Provisioning</a></li> </ul>

Tenant Type	SAP BTP Neo Environment	Infrastructure of SAP Cloud Identity	Details
<b>Standalone tenants</b> (Purchased between 01.09.2020 – 20.10.2020)	No	Yes	<p>Identity Provisioning and Identity Authentication are running on the same SAP Cloud Identity infrastructure.</p> <p><b>Access:</b> All regions and data centers where the Identity Authentication is running.</p> <p>See:</p> <ul style="list-style-type: none"> <li>• <a href="#">Identity Authentication: Regional Availability</a></li> <li>• <a href="#">SAP BTP Discovery Center: Identity Authentication</a></li> </ul>
<b>Standalone tenants</b> (After 20.10.2020)	No	No	<p>Identity Provisioning can no longer be purchased as a standalone product.</p>

## Related Information

[Updating Host URLs for Shanghai Tenants \[page 13\]](#)

### 1.1.3.1 Updating Host URLs for Shanghai Tenants

If your Identity Provisioning Neo tenants reside in the Asia-Pacific region, precisely Shanghai (China), you must use the new dedicated domain: **dispatcher.cn1.platform.sapcloud.cn**

## Context

The following domains of Identity Provisioning and Identity Authentication tenants located in Shanghai (China) will soon stop working:

- Domain of Identity Provisioning tenants: `dispatcher.cn1.hana.ondemand.com`
- Domain of Identity Authentication tenants: `accounts.ondemand.com`

## → Recommendation

We recommend that you start using the new domains and correct the URLs for all of your systems, where Identity Provisioning and Identity Authentication are involved by **September 30, 2021**. Use them as follows:

- New domain of Identity Provisioning tenants: `dispatcher.cn1.platform.sapcloud.cn`  
For example: `ips-<tenant_name>.dispatcher.cn1.platform.sapcloud.cn`
- New domain of Identity Authentication tenants: `accounts.sapcloud.cn`  
For example: `<tenant-id>.accounts.sapcloud.cn`

For more information, see: [Identity Authentication: Updating Host URLs for Shanghai Tenants](#)

This change affects the following scenarios:

- Identity Provisioning proxy systems
- Real-time provisioning configured and triggered in the Identity Authentication admin console
- Any other usage of Identity Authentication used as source or target system for user provisioning.

## Proxy Scenarios

If you have already configured connection to proxy systems in Identity Provisioning, you need to change the URL on the consumer side to reflect the new domain.

If you want to perform CRUD operations to an Identity Provisioning proxy system, make sure you use the correct domain when you construct your REST API requests.

For example: If your provider subaccount is **abcd12345**, your consumer account is **xyz789**, the ID of your proxy system in the Identity Provisioning UI is **bb111aa-1234-aaaa-7777-1234567abcde**, and you want to read a particular user (**s123456789**) from this system, then your REST API request should be:

```
GET https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/
ipsproxy/api/v1/scim/bb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789
```

To learn more, see: [Proxy Systems \[page 981\]](#) → section **How to call a proxy system**

## Real-Time Provisioning

If you want to perform real-time provisioning through the Identity Authentication admin console, make sure you use the correct domain when you construct your OAuth URL and SCIM URL. That means, your URLs should look like this:

- OAuth pattern:  
`https://oauthservices-<consumer_subaccount>.dispatcher.cn1.platform.sapcloud.cn/oauth2/api/v1/token`
- SCIM pattern:  
`https://ipsproxy<provider_account>-<consumer_subaccount>.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/systems/<Identity_Authentication_ID>/entities/user`

If you have already configured Identity Provisioning for real-time provisioning scenario through Identity Authentication, you need to change the URL on the consumer side to reflect the new domain. Also, when you access and refer to your Identity Authentication source system, use the new Identity Authentication domain: `https://<tenant-id>.accounts.sapcloud.cn/admin`

To learn more, see:

- [Real-Time Provisioning in SAP Cloud Identity Infrastructure \[page 1557\]](#)
- [Real-Time Provisioning in Neo Environment \[page 1560\]](#)

## Related Information

[Identity Provisioning: Regional Availability \[page 11\]](#)

[Identity Authentication: Regional Availability](#)

## 1.1.4 Disaster Recovery/High Availability

Disaster recovery (DR) and high availability (HA) are based on the capabilities of the underlying infrastructure.

SAP Cloud Identity Services – Identity Provisioning is a multi-tenant system where tenants share the hardware and software and use dedicated (and isolated) database instances for persistence.

### Disaster

A disaster is only declared by SAP when there is a loss of utilities and services and uncertainty on whether utilities and services can be restored within a reasonable period of time. As long as the production site has power and is connected to the Internet, it will not be considered a disaster.

Emergency incidents are assessed by SAP Business Technology Platform and SAP Corporate Infrastructure Services. An SAP management member with proper authorization must officially declare a disaster in order to initiate a disaster recovery plan.

Operations from the “disaster recovery site” could last anywhere from a few weeks to many months. Initiation of the failback plan is at SAP’s sole discretion.

### SAP BTP, Neo Environment

- Ensure your tenant is running on SAP BTP, Neo environment. For more information, see [Tenant Model \[page 8\]](#)

The Identity Provisioning service uses standard disaster recovery. Backups (complete data and log) are kept on a secondary location for the last **14 days**, and are deleted afterwards. For more information, see [Standard Disaster Recovery](#) and [Backup and Restore](#)

### Note

High availability and Multi-AZ are not supported for Identity Provisioning tenants running on SAP BTP, Neo environment. To take advantage of both features, you must migrate your tenant to SAP Cloud Identity Services infrastructure, as described in [Migrate Identity Provisioning Bundle Tenant \[page 1536\]](#).

Also, keep in mind that SAP BTP, Neo environment will sunset on December 31, 2028, subject to terms of customer or partner contracts. For more information, see [3351844](#).

## SAP Cloud Identity Infrastructure

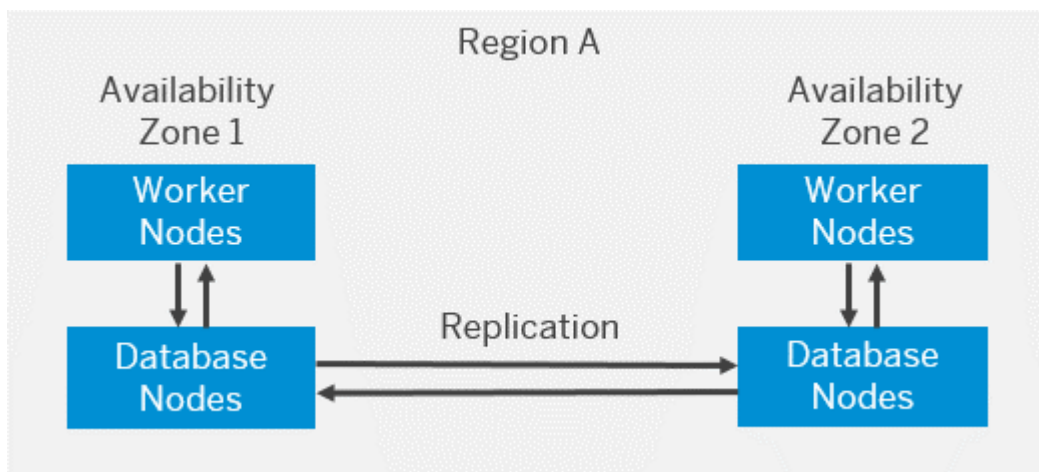
- Ensure your tenant is running on SAP Cloud Identity Services infrastructure. For more information, see [Tenant Model \[page 8\]](#)

Enhanced disaster recovery and high availability are fully supported for your tenants.

Disaster recovery and high availability are available only for the regions where Identity Authentication and Identity Provisioning share the same infrastructure and both services are enabled in a common tenant.

### High Availability – Single Region Setup

All deployments which have one data center support replication of the data between two zones within the same region.

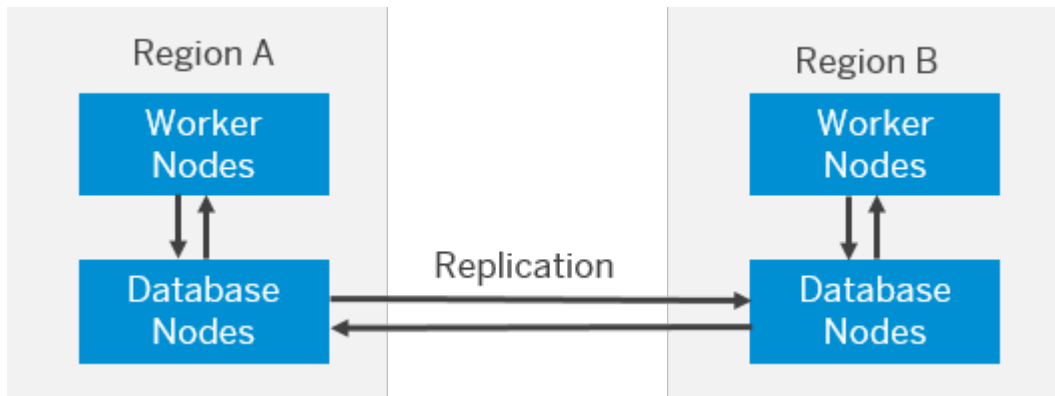


### High Availability/Disaster Recovery – Multi-Region Setup

Country/regions with two data centers operate in high availability (HA) and disaster recovery (DR) mode among the respective data centers. Tenants located in these country/regions are distributed among the data centers there.

Identity Provisioning uses Akamai GTM to route the traffic to a failover data center in case of any issues in the primary data center. This principle covers both the HA and DR setup.





## Related Information

[Identity Authentication: DR/HA](#)

### 1.1.5 Accessibility Features in Identity Provisioning

To optimize your experience of Identity Provisioning, the service provides features and settings that help you use the software efficiently.

#### **i** Note

Identity Provisioning is based on SAPUI5. For this reason, accessibility features for SAPUI5 also apply. See the accessibility documentation for SAPUI5 on SAP Help Portal at [Accessibility for End Users](#).

For more information on screen reader support and keyboard shortcuts, see [Keyboard Handling for SAPUI5 UI Elements](#) and [Screen-Reader Support for SAPUI5 Controls](#).


## 1.2 What's New for Identity Provisioning

[Archive \[page 63\]](#)

To check the latest release notes for the Identity Provisioning service, go to: [SAP Cloud Platform: What's New \(Identity Provisioning\)](#)

(Optional) You can change the default date filter (*From – To*) in order to see an extended or a narrowed range of release notes for the Identity Provisioning.

#### **→** Tip

Beside the official release notes, you can also see the road map of the Identity Provisioning to check the new features from the current and the upcoming quarter. See: [Identity Provisioning Road Map](#) 

## 19 December 2023 – Identity Provisioning

### Changed

#### Renamed System

The SAP S/4HANA Cloud for advanced financial closing connector has been renamed to SAP Advanced Financial Closing.

For more information, see: [SAP Advanced Financial Closing \[page 1028\]](#)

---

### Changed

#### Simplified error messages

The error messages are simplified and no longer display information about class name and package name.

For more information, see: [Monitor Provisioning Job Logs \[page 1594\]](#)

---

## 06 December 2023 – Identity Provisioning

### Changed

#### Automatic regeneration of outbound certificates

You can automatically regenerate and activate an outbound certificate within 14 days prior to its expiration. This way you will no longer need to periodically check your certificate validity and manually handle its renewal.

The functionality is available for customers with tenants running on SAP Cloud Identity Services infrastructure.

For more information, see: [Generate and Manage Certificates for Outbound Connection \[page 1507\]](#)

---

## 27 November 2023 – Identity Provisioning

### New

#### Manage transformations history

You can now manage the history of transformations, review and restore them to a previous version, as well as download a specific one. This way you will no longer need to copy, save and manage modified transformations in your own archive. Previously, you were able to only restore a transformation to its initial state.

Managing transformations history is supported for Identity Provisioning tenants running on SAP Cloud Identity infrastructure. Customers with tenants running on SAP BTP, Neo environment can only reset modified transformations to their initial state.

For more information, see:

- [Manage Transformations History \[page 1501\]](#)
- [Reset Identity Provisioning Transformations \[page 1504\]](#)

### Changed

#### Re-entering credentials required when changing URL or host

As of November 27, 2023, whenever you update the URL or the host name of a provisioning system, you must re-enter the values of the credential properties, such as `Password`, `ariba.applications.api.key` and others. The only exception to this are the credential properties of systems that are created with a connectivity destination.

For more information, see: [Manage Properties \[page 1505\]](#)

## 08 November 2023 – Identity Provisioning

### New

#### SAP Initiated system type introduced

SAP Initiated system type is introduced to represent provisioning systems that are automatically created and preconfigured in the Identity Provisioning UI. It comes in addition to the Customer Managed type which represents provisioning systems that you as a customer create and configure in the Identity Provisioning UI.

For more information, see: [System Types \[page 86\]](#)

### New

#### Download logs from notification e-mails

You can now download error logs and skipped entities logs directly from the notification e-mails that provide information about the job execution. Sending navigation links to the download page of failed or skipped entities is supported for tenants running on SAP Cloud Identity Services infrastructure.

For more information, see: [Manage Job Notifications \[page 1605\]](#)

## 02 November 2023 – Identity Provisioning

### New

#### isValidEmail function introduced

You can now use the isValidEmail function to verify whether an e-mail address is valid. It is used only within conditions and checks if the given string matches a particular regex pattern.

For more information, see: [Transformation Functions \[page 362\]](#)

---

### Changed

#### Changing users and groups update mechanism

Identity Provisioning changed the way users and groups are updated in the target systems.

The service will first apply the mapping of the entity unique attribute (most often *userName* for users and *displayName* for groups) and then will resolve conflicts if the entity exists in the target system. Currently, Identity Provisioning resolves the entity prior to applying the mapping of its unique attribute.

When the new mechanism is introduced, if the given mapping contains a function (for example, for removing a substring or concatenating a prefix), the service will handle it correctly. The entity won't be skipped and will be properly updated.

Be aware that this change will affect you if you have the following three configurations in place:

- The unique attribute is missing in the target transformation. Even if you've defined it as a property in `<system_prefix>.user.unique.attribute` or `<system_prefix>.group.unique.attribute`, it is still missing in the target transformation.
- The *skipOperation* create is defined under the user or group resource in the target transformation.
- The support for patch operations is enabled in the target system.

If you experience any issues when this change is introduced, make sure the unique attribute mapping is not missing in your target transformation. If this doesn't fix the issue, create a customer case to component BC-IAM-IPS. For more information, see [Getting Support \[page 1620\]](#)

---

### Changed

#### Groups are read in delta read mode

Reading groups in delta read mode is now supported for Identity Authentication version 2 and Local Identity Directory source systems.

For more information, see: [Manage Full and Delta Read \[page 1519\]](#)

---

### Changed

#### Local Identity Directory


In addition to standalone tenants, the Local Identity Directory connector is now enabled for all bundle tenants running on SAP Cloud Identity Services infrastructure. You can configure it as source, target and proxy system for your provisioning scenarios.

For more information, see: [Local Identity Directory \[page 465\]](#)

---

## Deprecated

### Root Certificate Replacement

DigiCert has deprecated their DigiCert Global Root CA and will stop issuing certificates for SAP under any of its Intermediate CAs (ICA) at the end of 2023. For more information, see [DigiCert root and intermediate CA certificate updates 2023](#) .

- If your tenant is running on SAP BTP, Neo environment and you are using Identity Provisioning proxy and real-time provisioning scenarios, ensure that you trust the new root CA: DigiCert Global Root G2. It replaces the old one DigiCert Global Root CA.  
For more information, see [Root Certificate Replacement](#).
- If your tenant is running on SAP Cloud Identity Services infrastructure on the domains listed below, ensure that you trust the new root CA: DigiCert Global Root G2 starting in mid-November 2023:
  - \*.accounts.ondemand.com
  - \*.accounts.cloud.sap
  - \*.accounts.sapcloud.cn
  - \*.trial-accounts.ondemand.com

For more information, see [Root Certificate Replacement](#) in *What's New for Identity Authentication*.

## 11 October 2023 – Identity Provisioning

## New

### In-App Help implemented

SAP Companion context-sensitive in-app help has been implemented in the SAP Cloud Identity Services administration console. You can start the in-app help by selecting the ⓘ [Help](#) control. The administration console provides [Help Topics](#), [Guided Tours](#), and [What's New](#) content.

For more information, see [SAP Companion User Guide](#).

## Changed

### Support for reading userName of group members

Reading the userName of group members is now supported in the default transformations of SAP S/4HANA-based source and proxy systems. This change affects SAP S/4HANA Cloud, SAP IBP, SAP Market Communication, SAP Marketing Cloud, and SAP BTP ABAP environment.

For more information, see [SAP S/4HANA Cloud \[page 613\]](#).

## Changed

### Identity Provisioning is not bundled with SAP Landscape Management Cloud

Identity Provisioning no longer bundles with SAP Landscape Management Cloud because the solution is no longer dependent on SAP Analytics Cloud, embedded edition. Previously, the service was involved in provisioning of users and groups from Identity Authentication source system to SAP Analytics Cloud target system.

As of now, SAP Landscape Management Cloud bundles with Identity Authentication only.

---

## 27 September 2023 – Identity Provisioning

## Announcement

### Changing users and groups update mechanism

As of October 25, 2023, Identity Provisioning is changing the way users and groups are updated in the target systems.

The service will first apply the mapping of the entity unique attribute (most often *userName* for users and *displayName* for groups) and then will resolve conflicts if the entity exists in the target system. Currently, Identity Provisioning resolves the entity prior to applying the mapping of its unique attribute.

When the new mechanism is introduced, if the given mapping contains a function (for example, for removing a substring or concatenating a prefix), the service will handle it correctly. The entity won't be skipped and will be properly updated.

Be aware that this change will affect you if you have the following three configurations in place:

- The unique attribute is missing in the target transformation. Even if you've defined it as a property in `<system_prefix>.user.unique.attribute` or `<system_prefix>.group.unique.attribute`, it is still missing in the target transformation.
- The *skipOperation* create is defined under the user or group resource in the target transformation.
- The support for patch operations is enabled in the target system.

If you experience any issues when this change is introduced, make sure the unique attribute mapping is not missing in your target transformation. If this doesn't fix the issue, create a customer case to component BC-IAM-IPS. For more information, see [Getting Support \[page 1620\]](#)

---

## New

### SAP SuccessFactors version 2 - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP SuccessFactors version 2 using SAP SuccessFactors Workforce SCIM API.

For more information, see [List of Properties \[page 94\]](#) → `sf.group.prefix`.

---

## Changed

### Updated business partner roles for SAP S/4 HANA-based systems

The default transformations for SAP S/4 HANA-based systems have been updated to correspond to the currently supported business partner roles.

The update affects all SAP S/4 HANA-based source, target, and proxy systems: SAP S/4HANA Cloud, SAP S/4HANA on-premise, SAP IBP, SAP Market Communication, SAP Marketing Cloud, and SAP BTP ABAP environment.

For more information, see [SAP S/4HANA Cloud \[page 613\]](#)

---

## 29 August 2023 – Identity Provisioning

## New

### SAP SuccessFactors version 2 - filtering of inactive users

You can now use `sf.user.read.deactivatedafter` to filter SAP SuccessFactors inactive users from a particular date on.

The property is supported for SAP SuccessFactors version 2 using SAP SuccessFactors Workforce SCIM API when configured as a source or proxy system. It works together with the `sf.user.filter` property which is added at system creation.

For more information, see [List of Properties \[page 94\]](#) → `sf.user.read.deactivatedafter`

---

## New

### Real-time provisioning of groups

You can now configure real-time provisioning for newly created or updated groups. Real-time provisioning of groups can be configured for each source system that supports the execution of requests to the `/Groups` endpoint of the Real-time provisioning API.

For more information, see [Real-Time Provisioning \[page 1554\]](#).

---

## 16 August 2023 – Identity Provisioning

### Changed

#### Enhanced group unassignment in target systems

You can now remove the group assignments of a deleted user from the source system. For this, you need to modify the write transformation by adding a mapping under the user resource and providing the group IDs. Thus, the user is persisted in the target system and instead only its status is updated.

This feature is supported for tenants running on SAP Cloud Identity Services infrastructure.

For more information, see: [Enabling Group Assignment \[page 1499\]](#)

---

## 02 August 2023 – Identity Provisioning

### New

#### copyMapEntry and renameMapEntry functions are introduced

You can now configure the `copyMapEntry` and the `renameMapEntry` functions in the Identity Provisioning transformations to copy and rename sub-attributes, respectively.

For more information, see [Transformation Functions \[page 362\]](#) → `copyMapEntry` and `renameMapEntry`

---

### New

SAP SuccessFactors connector version 2 (using SAP SuccessFactors Workforce SCIM API) is enhanced to support the provisioning of static permission groups and user's group assignments when configured as a proxy system.

For more information, see: [SAP SuccessFactors version 2 - provisioning of static groups in proxySAP SuccessFactors \(Proxy\) \[page 1360\]](#)

---

## 24 July 2023 – Identity Provisioning

### New

#### Download job logs via API

In addition to downloading job logs from the UI, you can now download them via API calls. This feature is supported for tenants running on SAP Cloud Identity Services infrastructure. Administrators must have the [Access Identity Provisioning Tenant Admin API](#) permission enabled.

For more information, see [Manage Provisioning Job Logs \[page 1600\]](#)

---



## 17 July 2023 – Identity Provisioning

### New

#### SAP SuccessFactors version 2 - provisioning of static groups in target systems

SAP SuccessFactors connector version 2 (using SAP SuccessFactors Workforce SCIM API) is enhanced to support the provisioning of static permission groups and user's group assignments when configured as a target system.

For more information, see: [SAP SuccessFactors \(Target\) \[page 919\]](#)

---

### Changed

#### SAP Concur using User Provisioning Service v4 API - authentication mechanism

SAP Concur connector version 1 (using UPS v4 API) adopts the authentication mechanism of SAP Concur connector version 2 (using Identity v4 API), that is [OAuth 2.0](#).

For more information, see:

- [SAP Concur \(Source\) \[page 548\]](#)
  - [SAP Concur \(Target\) \[page 817\]](#)
  - [SAP Concur \(Proxy\) \[page 1201\]](#)
- 

### Changed

#### SAP Commerce Cloud - group mapping

The SAP Commerce Cloud write transformation is changed to support the mapping of group **name** attribute to group **displayName** attribute. This ensures that the conflict resolution works properly in cases when the **name** is the unique group attribute in the source system instead of the **displayName** (for example, in Identity Authentication).

For more information, see [SAP Commerce Cloud \[page 808\]](#).

---

## 04 July 2023 – Identity Provisioning

### New

#### Real-time provisioning logs

In addition to job logs, you can now monitor real-time provisioning logs in the SAP Cloud Identity Services administration console. Real-time provisioning logs display information about the real-time sync execution of a user entity. They are supported for tenants running on SAP Cloud Identity Services infrastructure.

For more information, see [Monitor Real-Time Logs \[page 1598\]](#)

---

## New

### SAP Advanced Workflow connector

Identity Provisioning supports SAP Advanced Workflow connector. It is enabled for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, and bundle tenants running on SAP Cloud Identity infrastructure.

You can configure it as source, target and proxy system for your provisioning scenarios. For more information, see:

- [SAP Advanced Workflow \(Source\) \[page 475\]](#)
- [SAP Advanced Workflow \(Target\) \[page 728\]](#)
- [SAP Advanced Workflow \(Proxy\) \[page 1038\]](#)

## New

### Conflict resolution supported for SAP S/4 HANA-based systems

Properties defining unique attribute(s) for conflict resolution of existing users can now be configured for SAP S/4HANA Cloud, SAP S/4HANA on-premise, SAP IBP, SAP Market Communication, SAP Marketing Cloud and SAP BTP ABAP environment target and proxy systems.

The properties accept one of the following values: `personExternalId` (default value) and `email`.

For more information, see [List of Properties \[page 94\]](#) → `<system_prefix>.user.unique.attribute`, for example: `<s4hana.cloud>.user.unique.attribute`.

## New

### Multiple origins supported in proxy systems

You can configure and provision multiple origins for users in SAP BTP XS Advanced UAA (Cloud Foundry) proxy systems. Initially, multiple origins were supported for source and target systems only.

For more information, see [Configure Single and Multiple Origins \[page 1140\]](#).

## Changed

### Tenant migration - URL change

Following a successful migration of Identity Provisioning tenant from SAP BTP, Neo environment to SAP Cloud Identity Services infrastructure, your tenant will be available at the `<ias-host>/admin` URL.

The `<ias-host>/ips` URL is still in use for backwards compatibility. It redirects you to the SAP Cloud Identity Services administration console.

For more information, see [Migrate Identity Provisioning Bundle Tenant \[page 1536\]](#)

## Changed

### **replaceFirstString function used instead of replaceString**

The `replaceFirstString` function is used instead of the `replaceString` function in all default write transformations supporting group prefixes. This function replaces the first substring of a given string that matches the provided regex (the value of the respective group prefix property) with the string in the replacement (an empty string).

As a result, when setting up the group prefix property in a target system, for example `c4c.group.prefix=C4C`, groups will be provisioned as `C4C_<Display_Name>` instead of `C4C_C4C_<Display_Name>`.

For more information, see [replaceFirstString](#)

---

## 19 June 2023 – Identity Provisioning

## New

### **Controlled deletion of entities**

You can control the number of users, groups or roles to be deleted in a target system by defining a threshold. This will prevent you from accidentally deleting a huge number of entities, for example by adding a filter or condition. If the number of the entities to be deleted is greater than the defined threshold, Identity Provisioning will mark them as failed in the job statistics and will not delete them.

For more information, see [List of Properties \[page 94\]](#) → `ips.delete.threshold.users`, `ips.delete.threshold.groups` and `ips.delete.threshold.roles`.

---

## New

### **Microsoft Active Directory**

A new version of Microsoft Active Directory connector is introduced. It provides an improved performance of the read operation for user and group attributes. The new version allows you to read and preserve nested groups as the "group members" attribute mapping in the proxy read transformation is enhanced with "type" sub-attribute.

The `ldap.api.version` property differentiates the version of the connector.

For more information, see:

- [Microsoft Active Directory \(Source\) \[page 676\]](#)
  - [Microsoft Active Directory \(Target\) \[page 961\]](#)
  - [Microsoft Active Directory \(Proxy\) \[page 1436\]](#)
  - [List of Properties \[page 94\]](#) → `ldap.api.version`, `ldap.user.attributes` and `ldap.group.attributes`
-

## New

### SAP Field Service Management supports userUUID

SAP Field Service Management supports provisioning of users with the `userUUID` attribute. The default read, write and proxy read and write transformations have been enhanced to support the universally unique identifier.

For more information, see: [SAP Field Service Management \[page 575\]](#)

---

## Changed

### Renamed System

The SAP Market Communication connector has been renamed to SAP Market Communication for Utilities.

For more information, see: [SAP Market Communication for Utilities \[page 592\]](#)

---

## 06 June 2023 – Identity Provisioning

## New

### Multiple origins support

You can configure and provision multiple origins for users in SAP BTP XS Advanced UAA (Cloud Foundry). An origin tells you which is the identity provider of a user in SAP BTP XS Advanced UAA (Cloud Foundry). Its value is defined in the `xsuaa.origin` property.

For more information, see [Configure Single and Multiple Origins \[page 782\]](#) and [List of Properties \[page 94\]](#) → `xsuaa.origin`.

---

## Deprecated

### Local Identity Directory no longer supported on SAP BTP, Neo environment

The Local Identity Directory connector is no longer supported for Identity Provisioning tenants running on SAP BTP, Neo environment. You can use it in standalone tenants running on SAP Cloud Identity Services infrastructure.

For more information, see:

- [Local Identity Directory \(source\) \[page 465\]](#)
  - [Local Identity Directory \(target\) \[page 716\]](#)
  - [Local Identity Directory \(proxy\) \[page 1019\]](#)
- 

## New

### splitStringToArray function introduced

You can now configure the `splitStringToArray` function in the Identity Provisioning transformations to convert a sequence of strings into an array using a separator.

For more information, see [Transformation Functions \[page 362\]](#)

---

## New

### SAP S/4HANA Cloud - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP S/4HANA Cloud.

For more information, see [List of Properties \[page 94\]](#) → `s4hana.cloud.roles.prefix`.

---

## 25 May 2023 – Identity Provisioning

## New

### SAP Sales Cloud and SAP Service Cloud - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP Sales Cloud and SAP Service Cloud.

For more information, see [List of Properties \[page 94\]](#) → `c4c.group.prefix`.

---

## New

### LDAP Server

A new version of LDAP Server connector is introduced. It allows you to define which group and user attributes to be read from the external system by adding values to the properties `ldap.user.attributes` or `ldap.group.attributes`. With the new version you are able to read and preserve nested groups as the "group members" attribute mapping in the proxy read transformation is enhanced with "type" sub-attribute.

The `ldap.api.version` property differentiates the version of the connector.

For more information, see:

- [LDAP Server \(Source\) \[page 669\]](#)
  - [LDAP Server \(Target\) \[page 953\]](#)
  - [LDAP Server \(Proxy\) \[page 1423\]](#)
  - [List of Properties \[page 94\]](#) → `ldap.api.version`, `ldap.user.attributes` and `ldap.group.attributes`
-

## 10 May 2023 – Identity Provisioning

### New

#### SAP Cloud Identity Services admin console – redirects implemented

Customers accessing Identity Provisioning administration console at `http://<ias-host>/ips` tenant URL are now redirected to `https://<ias-host>/admin` and open the SAP Cloud Identity Services administration console. The entire provisioning functionality is embedded under *Identity Provisioning* section.

The redirect takes you to the *Source System* tile. If you've bookmarked a URL link pointing to a particular provisioning system or a tile, for example *Job Logs*, the redirect takes you there. This is implemented for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

For more information, see:

- [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
- [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

### New

#### OAuth2TokenScope property introduced

If your backend system is OAuth protected and requires an access token with scope, you can now use the `OAuth2TokenScope` property to specify the scope. It defines the token's level of access to protected resources.

For more information, see: [List of Properties \[page 94\]](#) → `OAuth2TokenScope`

## 24 April 2023 – Identity Provisioning

### New

#### SAP Advanced Financial Closing connector

Identity Provisioning supports SAP Advanced Financial Closing connector. It is enabled for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, and bundle tenants running on SAP Cloud Identity infrastructure.

You can configure it as source, target and proxy system for your provisioning scenarios. For more information, see:

- [SAP Advanced Financial Closing \(Source\) \[page 470\]](#)
- [SAP Advanced Financial Closing \(Target\) \[page 723\]](#)
- [SAP Advanced Financial Closing \(Proxy\) \[page 1028\]](#)

#### New

##### Microsoft Azure AD - support for advanced query filters

Identity Provisioning supports the advanced query filters implemented in Microsoft Graph API for Microsoft Azure AD users and groups. You can now filter out entities using operators supported in advanced queries, such as **ne** (not equal), as described in: [Advanced query capabilities on Azure AD objects](#) 🖱️.

For more information, see: [List of Properties \[page 94\]](#) → `aad.user.filter` and `aad.group.filter`.

---

## 13 April 2023 – Identity Provisioning

#### New

##### SAP Cloud Identity Services trial tenant

You can now create an SAP Cloud Identity Services trial tenant from an SAP BTP trial account. A trial tenant is intended for testing purposes of SAP Cloud Identity Services – Identity Authentication and Identity Provisioning.

For more information, see: [Getting a Trial Tenant \[page 411\]](#)

---

## 10 April 2023 – Identity Provisioning

#### New

##### SAP Data Custodian connector

Identity Provisioning supports SAP Data Custodian connector. It is enabled for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, and bundle tenants running on SAP Cloud Identity infrastructure.

You can configure the Software as a Service solution as source, target and proxy system for your provisioning scenarios.

For more information, see:

- [SAP Data Custodian \(Source\) \[page 561\]](#)
  - [SAP Data Custodian \(Target\) \[page 833\]](#)
  - [SAP Data Custodian \(Proxy\) \[page 1228\]](#)
-

## New

### SAP Analytics Cloud

A new version of SAP Analytics Cloud connector is introduced. It is based on the SAP Analytics Cloud SCIM API. Compared to the existing SCIM API version 1, the new version 2 is enhanced to support the patch operation in target and proxy systems. Also, reading groups is not ignored by default, as it is with version 1.

The `sac.api.version` property differentiates the version of the connector. The default value is: **1**.

For more information, see:

- [SAP Analytics Cloud \(Source\) \[page 478\]](#)
- [SAP Analytics Cloud \(Target\) \[page 731\]](#)
- [SAP Analytics Cloud \(Proxy\) \[page 1045\]](#)
- [Update Connector Version \[page 1484\]](#)

## 27 March 2023 – Identity Provisioning

## New

### phoneNumber supported as unique attribute for conflict resolution

The `phoneNumber` is supported as a unique attribute for conflict resolution of existing users in Identity Authentication (using SCIM API version 2) and Local Identity Directory (when Identity Provisioning is running on SAP Cloud Identity Infrastructure) target systems.

You can define the `phoneNumber` in the respective property: `ias.user.unique.attribute` (for Identity Authentication) and `scim.user.unique.attribute` (for Local Identity Directory) along with other supported unique attributes. The default value of these properties remains `userName`.

For more information, see: [List of Properties \[page 94\]](#) → `ias.user.unique.attribute` and `scim.user.unique.attribute`



## 16 March 2023 – Identity Provisioning

### New

#### Graphical editor introduced

Identity Provisioning provides a graphical editor for managing provisioning system transformations. It comes with a number of advantages over the current JSON text editor in terms of visualization, simplicity of use and validation. It brings improved user experience, requires less typing and more choosing from a list of prefilled values.

The graphical editor is available only for Identity Provisioning tenants running on SAP Cloud Identity infrastructure. It is the default editor.

For more information, see:

- [Transformation Editors \[page 402\]](#)
- [Working with Graphical Editor \[page 1495\]](#)

## 14 March 2023 – Identity Provisioning

### Changed

#### skipOperations - update of existing users

Identity Provisioning service can now execute update-only operations for existing entities in a target system while using `skipOperations` for create and delete. This behavior is achieved by using PATCH operation support for the corresponding target system. Thus, you are able to specify which attributes to be updated. The service resolves the user by the default or predefined unique attributes for the system.

Up to now, when Identity Provisioning executes the create operations, the existing entities are resolved by their resource IDs.

For more information, see [Transformation Expressions \[page 330\]](#) → `skipOperations`.

### New

#### SAP Marketing Cloud - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP Marketing Cloud connector.

For more information, see [List of Properties \[page 94\]](#) → `marketing.cloud.roles.prefix`.

### New

#### SAP Market Communication - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP Market Communication connector.

For more information, see [List of Properties \[page 94\]](#) → `maco.roles.prefix`.

## New

### SAP Integrated Business Planning for Supply Chain - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP Integrated Business Planning for Supply Chain.

For more information, see [List of Properties \[page 94\]](#) → `ibp.roles.prefix`.

---

## 28 February 2023 – Identity Provisioning

## New

### SAP AS ABAP - time zone property

You can now use the `abap.host.timezone` property to specify the time zone of your SAP AS ABAP on-premise systems in **UTC+/- offset** format. It is used for calculating the correct assignments validity in case your SAP AS ABAP system and Identity Provisioning tenant are running in different time zones.

For more information, see [List of Properties \[page 94\]](#) → `abap.host.timezone`.

---

## New

### SAP Jam Collaboration - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP Jam Collaboration connector.

For more information, see [List of Properties \[page 94\]](#) → `jam.group.prefix`.

---

## 14 February 2023 – Identity Provisioning

### New

#### Identity Provisioning integrated in SAP Cloud Identity Services admin console

You can now configure and work with Identity Provisioning in the administration console of SAP Cloud Identity Services (formerly known as the administration console of Identity Authentication).

The entire provisioning functionality, which includes adding, enabling, disabling, deleting and resetting provisioning systems, running jobs, viewing and downloading logs, is integrated there and can be accessed in the navigation area under *Identity Provisioning*.

The latest step in tightening SAP Cloud Identity Services integration allows you to manage your configurations in one place without the need to switch between consoles. To benefit from it, your Identity Provisioning tenant must run on SAP Cloud Identity Services infrastructure.


For more information, see [Configure Identity Provisioning in SAP Cloud Identity Services Administration Console \[page 1535\]](#).

### New

#### SAP Business Network connector

Identity Provisioning supports SAP Business Network (formerly known as SAP Ariba Network) connector. It is enabled for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, and bundle tenants running on SAP Cloud Identity infrastructure.

#### i Note

Currently, SAP Business Network connector is only available for selected customers who are approached by SAP. For more information, see [3305074](#) 

For more information, see:

- [SAP Business Network \(Source\) \[page 531\]](#)
- [SAP Business Network \(Target\) \[page 799\]](#)
- [SAP Business Network \(Proxy\) \[page 1164\]](#)

### New

#### Bulk support for SAP Build Work Zone, standard edition target systems

Bulk operations are now supported for SAP Build Work Zone, standard edition target systems. When bulk operations are enabled, Identity Provisioning creates, updates, and deletes multiple users or groups in one request. The maximum number of operations to be performed in one bulk request is 100.

For more information, see: [List of Properties \[page 94\]](#) → `cflp.support.bulk.operation`,  
`cflp.bulk.operations.max.count`

## New

### Search for job logs

You can search for provisioning job logs in the [Job Execution Logs](#) screen. The search field allows you to filter logs by system name, job type and status. This functionality is available for customers with Identity Provisioning tenants running on SAP Cloud Identity Services infrastructure.

For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#)

---

## 31 January 2023 – Identity Provisioning

## New

### SAP Enterprise Portal connector

Identity Provisioning supports SAP Enterprise Portal connector. It is enabled for all standalone and bundle tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment.

You can configure SAP Enterprise Portal only as a source system. For more information, see [SAP Enterprise Portal \[page 566\]](#).

---

## New

### Enable group assignments

You can assign users to groups through the user resource of SCIM-based target systems. For this, you need to modify the write transformation by defining a condition for the users and providing the group IDs. For more information, see [Enabling Group Assignment \[page 1499\]](#).

---

## New

### Create a system for specific version

When creating a provisioning system that supports more than one version (such as Identity Authentication, SAP SuccessFactors and SAP Concur), you can choose a version that differs from the default one. This way you create the system with the version specific properties and transformations.

For more information, see [Add a System \[page 1477\]](#)

---

## New

### Microsoft Azure AD - expand user and group attributes

You can now expand the list of user and group attributes which Identity Provisioning reads from Microsoft Azure Active Directory with additional attributes configured in the `aad.user.attributes.expand` and `aad.group.attributes.expand` properties, respectively.

For more information, see [List of Properties \[page 94\]](#) → `aad.user.attributes.expand` and `aad.group.attributes.expand`.

---

## Changed

### Renamed Systems

The following Identity Provisioning connectors have been renamed:

- SAP Work Zone → [SAP Build Work Zone, advanced edition](#)
- SAP Launchpad → [SAP Build Work Zone, standard edition](#)
- SAP S/4HANA Procurement Planning → [SAP S/4 HANA for procurement planning](#)

## 18 January 2023 – Identity Provisioning

### New

#### SAP SuccessFactors target system available in bundle tenants

You can configure SAP SuccessFactors as a target system in Identity Provisioning bundle tenants running on SAP Cloud Identity Services infrastructure. Previously, only SAP SuccessFactors source and proxy systems were available.

For more information, see: [SAP SuccessFactors \(Target\) \[page 919\]](#) and [SAP SuccessFactors Bundle \[page 438\]](#)

### New

#### Patch requests to target systems

Properties controlling how modified entities in the source system are updated in the target system: `scim.support.patch.operation`, `ias.support.patch.operation`, `ariba.applications.support.patch.operation` and others, can be configured on target systems. Previously, these properties were supported only for proxy systems.

When set to **true**, Identity Provisioning sends a patch request to the user or group resource in the target system. Only attributes without scope in the attribute mappings will be updated. This behavior is valid for all SCIM-based target systems.

For more information, see: [List of Properties \[page 94\]](#) → `ias.support.patch.operation`, `scim.support.patch.operation` and `ariba.applications.support.patch.operation`

### New

#### Read mode displayed in job execution details

The read mode of a provisioning job, that is: *Full Read* or *Delta Read*, is displayed in the job execution details screen.

For more information, see: [Monitor Provisioning Job Logs \[page 1594\]](#)

## New

### SAP CPQ supports userUUID

SAP CPQ supports provisioning of users with the `userUUID` attribute. The default read, write and proxy read and write transformations have been enhanced to support the universally unique identifier.

For more information, see: [SAP CPQ \[page 557\]](#)

---

## 20 December 2022 – Identity Provisioning

## New

### SAP Commerce Cloud connector

Identity Provisioning supports SAP Commerce Cloud connector. It is enabled for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, and bundle tenants running on SAP Cloud Identity infrastructure.

You can configure the cloud-based solution as source, target and proxy system for your provisioning scenarios. For more information, see:

- [SAP Commerce Cloud \(Source\) \[page 539\]](#)
  - [SAP Commerce Cloud \(Target\) \[page 808\]](#)
  - [SAP Commerce Cloud \(Proxy\) \[page 1181\]](#)
- 

## New

### Transformation functions `convertCountryCode` and `convertCountryRegion` are implemented

The following transformation functions: `convertCountryCode` and `convertCountryRegion` are implemented to convert countries into a format compliant with *ISO 3166-1* and *ISO 3166-2*, respectively.

For more information, see [Transformation Functions \[page 362\]](#) → `convertCountryCode` and `convertCountryRegion`.

---

## New

### SAP Concur - bundle option

SAP Concur is now bundled with SAP Cloud Identity Services – Identity Authentication and Identity Provisioning.

For more information, see: [SAP Concur Bundle \[page 447\]](#)

---

## New

### SAP Fieldglass - bulk support for groups' user members

Bulk operations are now supported for the user members of SAP Fieldglass groups. When bulk operations are enabled, Identity Provisioning updates multiple group members in a single request. The maximum number of operations to be performed in one bulk request is 100.

For more information, see: [List of Properties \[page 94\]](#) → `fg.support.bulk.operation`, `fg.bulk.operations.max.count`

---

## New

### SAP CPQ - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP CPQ connector.

For more information, see [List of Properties \[page 94\]](#) → `cpq.group.prefix`.

---

## 05 December 2022 – Identity Provisioning

## New

### SAP SuccessFactors – Support for X.509 certificate-based authentication

SAP SuccessFactors supports X.509 certificate-based authentication for incoming calls from Identity Provisioning. Certificate-based authentication using mutual Transport Layer Security (mTLS) provides a more secure authentication option to its users.

For more information, see:

- [X.509 Certificate-Based Authentication for Incoming Calls](#)
  - [Generate and Manage Certificates for Outbound Connection \[page 1507\]](#)
- 

## New

### SAP SuccessFactors connector based on SCIM API

A new version of SAP SuccessFactors connector is introduced. It is based on the SAP SuccessFactors Workforce SCIM API. Previously, the connector was based on SAP SuccessFactors HCM Suite OData API.

The `sf.api.version` property differentiates which API you use.

For more information, see:

- [New SCIM APIs for SAP SuccessFactors Workforce Users](#)
  - [List of Properties \[page 94\]](#) → `sf.api.version`.
-

## New

### SAP Market Communication connector

Identity Provisioning supports SAP Market Communication connector. It is enabled for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, and bundle tenants running on SAP Cloud Identity infrastructure.

The SAP Market Communication application is based on SAP BTP ABAP environment. You can configure it as source, target, and proxy system for your provisioning scenarios. For more information, see:

- [SAP Market Communication \(Source\) \[page 592\]](#)
- [SAP Market Communication \(Target\) \[page 873\]](#)
- [SAP Market Communication \(Proxy\) \[page 1281\]](#)

## New

### SAP Fieldglass - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP Fieldglass connector.

For more information, see [List of Properties \[page 94\]](#) → `fg.group.prefix`.

## 21 November 2022 – Identity Provisioning

## Changed

### CloudConnectorLocationId - supported for all HTTP and LDAP-based systems

The Identity Provisioning `CloudConnectorLocationId` property can now be configured for all HTTP and LDAP-based provisioning systems which have `ProxyType` set to **OnPremise**. This property holds the Location ID which identifies the Cloud Connector over which the connection to on-premise systems is opened. Previously, it was supported for SSH Server (Beta) and SAP HANA Database (Beta) only.

For more information, see: [List of Properties \[page 94\]](#) → `CloudConnectorLocationId`.

## Changed

### SAP Sales Cloud and SAP Service Cloud - support for reading group assignments through user resource

The proxy read transformation of SAP Sales Cloud and SAP Service Cloud (formerly known as SAP Cloud for Customer) supports reading of group assignments through the user resource.

For more information, see: [SAP Sales Cloud and SAP Service Cloud \[page 1350\]](#).



## 08 November 2022 – Identity Provisioning

### New

#### Validate provisioning job

You can validate a provisioning job before you actually run it. Validating a job allows you to test how entities (users and groups) would be mapped from source to target systems. For example, whether an attribute is required or optional, and whether a condition is fulfilled. Like the simulate job, the validate one does not modify the target system.

To run a validate job you need to create and import one or two CSV files - one for testing users and/or one for testing groups. The result is not provided in the Job Logs but in a zipped file, containing a number of CSV files.

For more information, see [Validate Provisioning Jobs \[page 1529\]](#)

---

### New

#### SAP Fieldglass - bulk support

Bulk operations are now supported for SAP Fieldglass target systems. When bulk operations are enabled, Identity Provisioning creates, updates, and deletes multiple users in one request. The maximum number of operations to be performed in one bulk request is 100.

For more information, see: [List of Properties \[page 94\]](#) → *fg.support.bulk.operation*, *fg.bulk.operations.max.count*

---

### New

#### User UUID callback to SAP SuccessFactors and SAP SuccessFactors Learning supported in bulk scenario

Returning the user UUID from Identity Authentication back to SAP SuccessFactors and SAP SuccessFactors Learning is now supported in bulk scenario. This means that when bulk operations are enabled on the Identity Authentication target system, the generated value of the user UUID will be sent back to those systems and will update the respective attributes there.

For more information, see [SAP SuccessFactors \[page 635\]](#) and [SAP SuccessFactors Learning \[page 649\]](#).

---

### Changed

#### Horizon theme switched on

The Horizon theme of SAP Fiori is now switched on by default for the Identity Provisioning user interface. It is supported for bundle and standalone tenants running on both - SAP Cloud Identity infrastructure and SAP BTP, Neo environment.

The Horizon visual theme comes with a modern, friendly user interface, based on an accessible, modular design system. Compared to the previous Belize theme, it introduces the following changes: bold typography, rounded corners, a new icon for job logs and others.

For more information, see [SAP's UI Technologies supporting the new Horizon visual theme of SAP Fiori](#) 

---

## Changed

### System names with up to 100 characters

System names can have a length of up to 100 characters. Previously, up to 50 characters were allowed.

For more information, see [Add a System \[page 1477\]](#).

---

## 25 October 2022 – Identity Provisioning

## New

### SAP Central Business Configuration - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP Central Business Configuration connector.

For more information about the `cbc.group.prefix` property, see [List of Properties \[page 94\]](#)

---

## New

### SAP SuccessFactors Learning - new property

The `lms.instance.host` property is introduced for SAP SuccessFactors Learning. It holds the host of the SAP SuccessFactors Learning instance and must be configured when client certificate authentication is used for the communication between Identity Provisioning and SAP SuccessFactors Learning.

For more information about the `lms.instance.host` property, see [List of Properties \[page 94\]](#)

---

## Changed

### Local Identity Directory - patch requests

Configuring `is.scim.patched.entity` in the write transformation of the Local Identity Directory is no longer needed for executing patch requests.

For more information, see [Patched and Merged Attributes \[page 1590\]](#)

---

## 10 October 2022 – Identity Provisioning

### New

#### SAP BTP bundle - create Identity Authentication and Identity Provisioning test tenant

You can now create an Identity Authentication and Identity Provisioning test tenant for SAP BTP, Cloud Foundry environment, in SAP BTP cockpit. Previously, only a productive tenant was created this way, while the test tenant was created by opening an incident.

For more information, see [SAP Business Technology Platform Bundle \[page 427\]](#)

---

### New

#### SAP BTP XS Advanced UAA (Cloud Foundry) - implement group prefix

The group prefix mechanism that is used for distinguishing groups of a given provisioning system is now implemented for SAP BTP XS Advanced UAA (Cloud Foundry) connector.

For more information about the `xsuaa.group.prefix` property, see [List of Properties \[page 94\]](#)

---

### New

#### SAP Launchpad - define threshold of group members for patch requests

A property which defines the threshold number of group members above which they are provisioned on batches with PATCH requests is now implemented for SAP Launchpad connector. The default and maximum value is 5000.

For more information about the `cflp.patch.group.members.above.threshold` property, see [List of Properties \[page 94\]](#)

---

## 27 September 2022 – Identity Provisioning

### Changed

#### SAP Launchpad Service provider ID in proxy scenario

For SAP Launchpad Service proxy systems, the `providerId` of SAP Launchpad Service is handled by the transformations. Previously, it was only possible to define its value in the `cflp.providerId` property.

If you now provide it both in the transformations and the property, the value in the transformations has priority.

For more information, see: [SAP Build Work Zone, standard edition \[page 1154\]](#)

---

## Changed

### Changes in preconfigured systems for SAP S/4HANA Cloud bundle

The preconfigured SAP S/4HANA Cloud source system and the embedded SAP Analytics Cloud target system in Identity Provisioning bundle tenants for SAP S/4HANA Cloud release 2208 or higher contain the following changes compared to the initial preconfiguration that was first delivered with SAP S/4HANA Cloud release 1911:

- The condition for the user entity in the read transformation of SAP S/4HANA Cloud now ensures that users are replicated during their validity, not during the validity of the Employee role of their business partner.
- The `ips.delete.existedbefore.entities=true`, which ensures that deleted users in the source system are also deleted in the target one, is now preconfigured for SAP Analytics Cloud.

For more information, see: [SAP S/4HANA Cloud Bundle \[page 442\]](#)

---

## 14 September 2022 – Identity Provisioning

## Changed

### New e-mail address of Identity Provisioning notifications

As of September 14, 2022, Identity Provisioning changes the From address of its e-mail notifications. The service now sends e-mails from `ips@notifications.sap.com` instead of the currently used `noreply@sap.com` address.

The change affects only Identity Provisioning tenants running on SAP BTP, Neo environment.

Action: In case you apply rules to your e-mails, you need to adjust them accordingly.

---

## New

### SAP S/4HANA for Procurement Planning connector

Identity Provisioning supports SAP S/4HANA for Procurement Planning connector. It is enabled for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, and bundle tenants running on SAP Cloud Identity infrastructure.

You can configure the cloud-based solution as source, target and proxy system for your provisioning scenarios. For more information, see:

- [SAP S/4HANA for Procurement Planning \(Source\) \[page 622\]](#)
  - [SAP S/4HANA for Procurement Planning \(Target\) \[page 897\]](#)
  - [SAP S/4HANA for Procurement Planning \(Proxy\) \[page 1333\]](#)
- 

## New

### Search for source, target and proxy systems

You can now use a search field to filter out source, target and proxy systems by system name. The search field is available for Identity Provisioning tenants running on SAP Cloud Identity infrastructure.

For more information, see: [Search and Edit a System \[page 1480\]](#)

---

## 31 August 2022 – Identity Provisioning

### New

#### Group prefix implemented

Identity Provisioning service provides a mechanism to distinguish groups of a given provisioning system by specific prefix.

The group prefix is controlled both - by a property, where you provide the prefix value, and by the read and write transformations of source and target systems. It is currently supported for seven connectors through the corresponding properties:

- SAP BTP Java/HTML5 apps (Neo) - `hcp.group.prefix`
- SAP BTP Account Members (Neo) - `scp.group.prefix`
- SAP Application Server ABAP - `abap.role.prefix`
- SAP BTP ABAP environment - `a4c.roles.prefix`
- SAP Ariba Applications - `ariba.applications.group.prefix`
- SAP Field Service Management - `fsm.group.prefix`
- SAP Analytics Cloud - `sac.group.prefix`

For more information, see the documentation for the respective connector and the [List of Properties \[page 94\]](#)

---

## 16 August 2022 – Identity Provisioning

### New

#### SCIM System and LDAP Server enabled as source and proxy systems in bundle tenants

SCIM System and LDAP Server are now enabled as source and proxy systems in Identity Provisioning bundle tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment.

When configured in proxy scenarios, SCIM System and LDAP Server can be used only for reading of entities.

For more information, see:

- [Provisioning Systems for Bundle Tenants \[page 416\]](#)
  - [Bundle Tenants and Connectors \[page 422\]](#)
-

## New

### SAP Field Service Management connector

Identity Provisioning supports SAP Field Service Management connector. It is enabled for all standalone tenants and bundle tenants running on SAP Cloud Identity infrastructure.

You can configure the cloud-based solution as source, target and proxy system for your provisioning scenarios. For more information, see:

- [SAP Field Service Management \(Source\) \[page 575\]](#)
- [SAP Field Service Management \(Target\) \[page 848\]](#)
- [SAP Field Service Management \(Proxy\) \[page 1249\]](#)

## 11 July 2022 – Identity Provisioning

## New

### Run job through API

You can now run a provisioning job through API. To do this, you need a technical user with [Access Identity Provisioning Tenant Admin API](#) permission assigned. For more information, see [Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#)

The API is available on the

SAP Business Accelerator Hub: [SAP Cloud Identity Services](#) ➤ [Identity Provisioning Service](#) ➤ [API Reference](#) ➤ [Jobs](#) ➤.

## New

### Simulate provisioning job

You can simulate a provisioning job before you actually run it. Simulating a provisioning job allows you to test your Identity Provisioning configurations and see whether they produce the desired result in the target system.

For more information, see [Simulate Provisioning Jobs \[page 1528\]](#)

## New

### SAP SuccessFactors Learning – target and proxy system

You can configure SAP SuccessFactors Learning as a target and a proxy system in the Identity Provisioning admin console. Previously, the learning solution was available only as a source system.

SAP SuccessFactors Learning bundle option is extended with the newly supported target and proxy systems.

For more information, see:

- [SAP SuccessFactors Learning Bundle \[page 440\]](#)
  - [SAP SuccessFactors Learning \(Target\) \[page 934\]](#)
  - [SAP SuccessFactors Learning \(Proxy\) \[page 1383\]](#)
- 

## Changed

### SAP AS ABAP - write and proxy write transformations

The write and proxy write transformations of SAP AS ABAP no longer support managing of group assignments through the user resource. The user to group assignments are managed through the group resource using the "members" attribute.

For more information, see:

- [SAP Application Server ABAP \(Target\) \[page 741\]](#)
  - [SAP Application Server ABAP \(Proxy\) \[page 1059\]](#)
- 

## 27 June 2022 – Identity Provisioning

## New

### Migrating Identity Provisioning bundle tenants from Neo environment to SAP Cloud Identity infrastructure

Administrators of Identity Provisioning bundle tenants on SAP BTP, Neo environment can now migrate them to the infrastructure of SAP Cloud Identity Services.

Migrating bundle tenants to the infrastructure of SAP Cloud Identity Services improves the integration between the group of services that provide cloud identity capabilities: Identity Authentication, Identity Provisioning, and Identity Directory.

It allows you to take advantage of all Identity Provisioning new features, which from now on are released only for tenants on SAP Cloud Identity infrastructure.

For more information, see [Migrate Identity Provisioning Bundle Tenant \[page 1536\]](#)

---

## New

### Enabling Identity Provisioning connectors for bundle tenants on SAP Cloud Identity infrastructure

Administrators of bundle tenants running on SAP Cloud Identity infrastructure can now configure most of the provisioning systems (connectors) supported by Identity Provisioning for synchronizing user data.

There are a few exceptions, described here: [Bundle Tenants and Connectors \[page 422\]](#)

Bundle tenants running on SAP BTP, Neo environment still have a restricted set of allowed provisioning systems (except for SAP Cloud Identity Access Governance). To remove the restriction and enable the usage of the supported Identity Provisioning connectors, you need to migrate your bundle tenant on Neo environment to the SAP Cloud Identity infrastructure. For more information, see [Migrate Identity Provisioning Bundle Tenant \[page 1536\]](#)

---

## New

### Download all skipped entities for a provisioning job

You can download and view the details of all skipped entities for a given provisioning job.

This feature is supported for bundle and standalone tenants running on SAP Cloud Identity infrastructure.

For more information, see [Manage Provisioning Job Logs \[page 1600\]](#)

---

## New

### Run job on a specific day of the week and time

When scheduling a job, you can now specify the day of the week and the exact time to run the job. Previously, you were only able to schedule (in minutes) how often a job to be run.

This feature is supported for bundle and standalone tenants running on SAP Cloud Identity infrastructure.

For more information, see [Start and Stop Provisioning Jobs \[page 1524\]](#)

---

## New

### Color code and autocompletion enabled for transformations

Color code and autocompletion are enabled by default for transformations in all source, target and proxy systems.

This feature is supported for bundle and standalone tenants running on SAP Cloud Identity infrastructure.

---

## New

### Navigate between Identity Authentication and Identity Provisioning

Administrators of Identity Authentication and Identity Provisioning tenants running on SAP Cloud Identity infrastructure can directly navigate between their administration consoles.

For more information, see [Configure Identity Provisioning in SAP Cloud Identity Services Administration Console \[page 1535\]](#)

---



## New

### SAP SuccessFactors Learning - bundle option

SAP SuccessFactors Learning is now bundled with Identity Provisioning and Identity Authentication.

For more information, see: [SAP SuccessFactors Learning Bundle \[page 440\]](#)

---

## 26 May 2022 – Identity Provisioning

## Changed

### Resume and Pause buttons replaced with ON and OFF switch

The *Resume* and *Pause* buttons on the *Jobs* tab are replaced with *ON* and *OFF* switch. Unlike the buttons, the switch is not displayed on the *Jobs* tab of the source systems. To use it, you need to choose *Schedule* for read jobs and turn it on or off in the *Job Scheduler* screen.

For more information, see [Start and Stop Provisioning Jobs \[page 1524\]](#)

---

## 17 May 2022 – Identity Provisioning

## New

### Client Certificates for Inbound Connection

Identity Provisioning supports client certificates for inbound connection.

In inbound connections, Identity Provisioning acts as a server whereas the given provisioning system acts as a client and must present a client certificate for establishing the communication to the service. Inbound certificates are supported for source and proxy systems in the following scenarios: configuring proxy systems and real-time provisioning.

For more information, see [Manage Certificates \[page 1506\]](#) and [Manage Certificates for Inbound Connection \[page 1510\]](#)

---

## 27 April 2022 – Identity Provisioning

### New

#### SAP Ariba Applications nested groups

You can now update only user members of a group in SAP Ariba Applications target system via PATCH request. This will preserve a group hierarchy with nested groups in the SAP Ariba Applications backend.

The maximum number of user members of a group that can be included in one PATCH request is 200 000.

For more information, see `ariba.applications.patch.group.members.of.nested.groups` and `ariba.applications.patch.group.members.above.threshold` in [List of Properties \[page 94\]](#).

---

## 07 April 2022 – Identity Provisioning

### New

#### SAP Work Zone bundle

SAP Work Zone is now bundled with Identity Provisioning and Identity Authentication.

To obtain an Identity Provisioning tenant, you need to connect your subaccount to Identity Provisioning in Work Zone Manager as described in [SAP Build Work Zone, advanced edition Bundle \[page 444\]](#).

---

## 29 March 2022 – Identity Provisioning

### New

#### Manage the lifecycle of two certificates

Identity Provisioning administrators can manage up to two certificates for the secure communication with a given provisioning system. Generating a second certificate might be needed when your active certificate approaches its expiration date.

For more information on how to manage certificates lifecycle, see [Generate and Manage Certificates for Outbound Connection \[page 1507\]](#)

---

## 15 March 2022 – Identity Provisioning

### New

#### New Identity Provisioning bundle tenants run on SAP Cloud Identity Services infrastructure

As of March 15, 2022, new Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. Existing customers of bundle tenants on Neo environment can continue using them as-is.

Delivering bundle tenants on the infrastructure of SAP Cloud Identity Services improves the integration between the group of services that provide cloud identity capabilities: Identity Authentication, Identity Provisioning, and Identity Directory.

For more information about the changes between Neo environment and SAP Cloud Identity Services infrastructure, see [Tenant Infrastructure \[page 10\]](#)

---

## 05 March 2022 – Identity Provisioning

### New

#### Conflict resolution for MS Active Directory and LDAP Server target systems

The **distinguishedName** value of the `ldap.attribute.dn` property is used for conflict resolution of existing users and groups in MS Active Directory and LDAP Server target systems.

This property defines by which unique attribute the entities will be searched and resolved. If an existing entity is found in MS Active Directory and LDAP Server target systems, Identity Provisioning will update it instead of creating a new one.

For more information, see: [List of Properties \[page 94\]](#) → `ldap.attribute.dn`

---

## 08 February 2022 – Identity Provisioning

### New

#### Proxy Systems tile enabled by default

The [Proxy Systems](#) tile is now enabled by default in the Identity Provisioning admin console. Existing and new customers no longer need to request access to the [Proxy Systems](#) tile by opening an incident.

---

## 11 January 2022 – Identity Provisioning

### New

#### Bulk support for Local Identity Directory target systems

Bulk operations are now supported for Local Identity Directory target systems (when Identity Provisioning and Identity Authentication are running on the same infrastructure, that is, the infrastructure of Identity Authentication). When bulk operations are enabled, Identity Provisioning creates, updates, and deletes multiple users or groups in one request. The maximum number of operations to be performed in one bulk request is 100.

For more information, see: [List of Properties \[page 94\]](#) → `idds.support.bulk.operation`, `idds.bulk.operations.max.count`

---

## 05 January 2022 – Identity Provisioning

### New

#### SAP Landscape Management Cloud - bundle option

SAP Landscape Management Cloud is now bundled with Identity Provisioning and Identity Authentication. Customers of SAP Landscape Management Cloud can use Identity Provisioning to synchronize users between Identity Authentication (source system) and SAP Analytics Cloud, embedded edition (target system).

Identity Provisioning does not provide a dedicated connector for SAP Landscape Management Cloud.

---

## 20 December 2021 – Identity Provisioning

### New

#### Support for certificate-based authentication

Identity Provisioning supports certificate-based authentication for secure communication with the provisioning systems (connectors) provided by the service.

Certificates can be used in outgoing connections where Identity Provisioning acts as a client. The service generates an X.509 client certificate for mutual Transport Layer Security (mTLS) authentication against a given provisioning system acting as a server.

For more information, see: [Manage Certificates \[page 1506\]](#)

---

## 08 December 2021 – Identity Provisioning

### New

#### SAP Concur

A new version of SAP Concur connector is introduced. It is based on the System for Cross-domain Identity Management (SCIM) API. Compared to the existing SAP Concur API, the new SAP Concur SCIM API supports the `userUUID` attribute which is generated by Identity Authentication at user creation.

The `concur.api.version` property differentiates which API you use. When set to **2**, the default SAP Concur SCIM API is used.

For more information, see:

- [SAP Concur \(Source\) \[page 548\]](#)
- [SAP Concur \(Target\) \[page 817\]](#)
- [SAP Concur \(Proxy\) \[page 1201\]](#)
- [Update Connector Version \[page 1484\]](#)

### New

#### Reset transformations

You can now reset Identity Provisioning system transformations to restore them to their initial state. Resetting transformations is supported for source, target, and proxy systems. You can only reset modified transformations.

For more information, see: [Reset Identity Provisioning Transformations \[page 1504\]](#)

## 30 November 2021 – Identity Provisioning

### New

#### UserUUID update in SAP SuccessFactors

Identity Provisioning can now return the `userUUID` from Identity Authentication to SAP SuccessFactors and update the `sapGlobalUserId` attribute if its value is not null. Previously, this was only possible if the attribute value in SAP SuccessFactors was null.

For more information, see: [SAP SuccessFactors \[page 635\]](#)

## 23 November 2021 – Identity Provisioning

### New

#### SAP Fieldglass - bundle option

SAP Fieldglass is now bundled with Identity Provisioning and Identity Authentication. To obtain an SAP Fieldglass bundle tenant, you need to create an incident.

For more information, see: [SAP Fieldglass Bundle \[page 448\]](#)

---

### New

#### Bulk support for Identity Authentication target systems based on SCIM API version 2

Bulk operations are now supported for Identity Authentication target systems which are based on SCIM API version 2. When bulk operations are enabled, Identity Provisioning creates, updates, and deletes multiple users or groups in one request. The maximum number of operations to be performed in one bulk request is 100.

For more information, see: [List of Properties \[page 94\]](#) → *ias.support.bulk.operation*, *ias.bulk.operations.max.count*

---

## 08 November 2021 – Identity Provisioning

### New

#### SAP SuccessFactors bundle extended with Identity Authentication source system

The scope of SAP SuccessFactors bundle option is now extended and includes Identity Authentication as a source system.

For more information, see: [SAP SuccessFactors Bundle \[page 438\]](#)

---

## 26 October 2021 – Identity Provisioning

### New

#### Query parameters include and exclude resource attributes in proxy SCIM API

When using Identity Provisioning proxy SCIM API, you can now specify the `attributes` and the `excludedAttributes` query parameters to control which user or group resource attributes to be included or excluded from the response.

These query parameters can be combined with other parameters, such as: filtering and paging of resources.

For more information, see: [Proxy Systems \[page 981\]](#) → *Query Parameters for Proxy SCIM API*

---

## 21 October 2021 – Identity Provisioning

### New

#### Patch updates above threshold number of group members

You can now define the threshold number of group members above which they will be provisioned on batches with PATCH requests. This functionality is supported for Identity Authentication, Local Identity Directory and SAP BTP XS Advanced UAA (Cloud Foundry) target systems by configuring a dedicated property for each of the systems.

The property allows you to avoid timeouts when updating groups with a large number of group members.

For more information, see: [List of Properties \[page 94\]](#) → *ias.patch.group.members.above.threshold*, *idds.patch.group.members.above.threshold* and *xsuaa.patch.group.members.above.threshold*

---

### New

#### SAP S/4HANA On-Premise supports provisioning of user UUID

SAP S/4HANA On-Premise supports provisioning of users with user UUID attribute which is generated by Identity Authentication at user creation. The attribute mapping is handled by the default transformation of SAP Application Server ABAP.

For more information, see: [SAP S/4HANA On-Premise \[page 625\]](#) and [SAP Application Server ABAP \[page 484\]](#)

---

## 23 September 2021 – Identity Provisioning

### Announcement

#### New domain of Identity Provisioning Neo tenants located in Shanghai (China)

The domain `dispatcher.cn1.hana.ondemand.com` of Identity Provisioning Neo tenants located in Shanghai (China) will soon stop working.

Therefore, we recommend that you start using the new domain `dispatcher.cn1.platform.sapcloud.cn` and correct the URLs for all of your systems, where Identity Provisioning is involved. For example: `ips-<tenant_name>.dispatcher.cn1.platform.sapcloud.cn`.

You must update to the new domain by September 30, 2021.

For more information, see: [Updating Host URLs for Shanghai Tenants \[page 13\]](#)

---

## New

### Support for UserUUID in communication scenario SAP\_COM\_0193

The communication scenario SAP\_COM\_0193 (*SAP Cloud Identity Provisioning Integration*) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

This scenario is used to enable the communication between Identity Provisioning and the following systems: SAP S/4HANA Cloud, SAP Marketing Cloud, SAP Integrated Business Planning for Supply Chain and SAP BTP ABAP environment.

For more information about the User UUID and the changes in the default transformations, refer to the respective documentation for each system (source, target and proxy):

- [SAP S/4HANA Cloud \[page 613\]](#)
- [SAP Marketing Cloud \[page 601\]](#)
- [SAP Integrated Business Planning for Supply Chain \[page 580\]](#)
- [SAP BTP ABAP environment \[page 502\]](#)

## New

### Identity Authentication - Automatic conflict resolution

When provisioning is triggered from source systems containing different users with the same user identifiers (IDs), you can use the `ias.user.automatic.conflict.resolution` property to control whether automatic conflict resolution is switched on or off in Identity Authentication (target system).

To learn more, see: [List of Properties \[page 94\]](#) → `ias.user.automatic.conflict.resolution`

## 27 August 2021 – Identity Provisioning

## New

### MS Azure AD – Combine user and group filtering

You can now use `aad.user.filter.group.filter.combine` property to filter Microsoft Azure Active Directory users based on their group assignments.

When set to **true**, this property combines user and group filters defined on the `aad.user.filter` and `aad.group.filter` properties to further narrow the search results. This way, only users that meet the following filtering criteria are returned:

- Users that match the user filter and at the same time are members of groups that match the group filter.
- Members of the filtered groups that match the user filter.

To learn more, see: [List of Properties \[page 94\]](#) → `aad.user.filter.group.filter.combine`



## 12 August 2021 – Identity Provisioning

### New

#### New target systems in all bundles

Along with source and proxy systems, you can now use the following connectors as targets as well:

- [SAP Application Server ABAP](#)
- [SAP S/4HANA On-Premise](#)

To learn more, see: [Provisioning Systems for Bundle Tenants \[page 416\]](#)

---

## 4 August 2021 – Identity Provisioning

### New

#### SAP IBP – bundle option

Identity Provisioning is now bundled with [SAP Integrated Business Planning for Supply Chain](#) (SAP IBP). You can request an Identity Provisioning bundle tenant (free of charge) by creating an incident. To learn more, see:

- [Obtain a Bundle Tenant \[page 407\]](#)
  - [SAP Integrated Business Planning for Supply Chain Bundle \[page 432\]](#)
  - [Provisioning Systems for Bundle Tenants \[page 416\]](#)
- 

## 29 July 2021 – Identity Provisioning

### New

#### SAP S/4HANA Cloud – bulk operations

You can now enable bulk operations to provision entities to SAP S/4HANA Cloud based target systems. Enabling bulk operations means that Identity Provisioning can write, update, and delete multiple users or groups in a single request. To do this, you need to use the following properties, depending on the relevant system type:

- `<system_prefix>.support.bulk.operation`
- `<system_prefix>.bulk.operations.max.count`

To learn more, see:

- [SAP S/4HANA On-Premise \(Target\) \[page 902\]](#)
  - [SAP S/4HANA Cloud \(Target\) \[page 888\]](#)
  - [SAP Marketing Cloud \(Target\) \[page 881\]](#)
  - [SAP BTP ABAP environment \(Target\) \[page 761\]](#)
  - [SAP Integrated Business Planning for Supply Chain \(Target\) \[page 860\]](#)
-

## 9 July 2021 – Identity Provisioning

### New

#### Identity Authentication

A new version of Identity Authentication connector is introduced. It is based on the Identity Directory SCIM API. Compared to the existing version that is based on the Identity Authentication SCIM API, the new version supports paging for group members and user's groups, patch operations for proxy systems, delta read for users and others. Reading and writing groups is no longer ignored in the default transformations.

To start using the new version, you need to follow an update procedure, described for Identity Authentication source, target and proxy systems. To learn more, see:

- [Identity Authentication \(Source\) \[page 453\]](#)
- [Identity Authentication \(Target\) \[page 702\]](#)
- [Identity Authentication \(Proxy\) \[page 998\]](#)

### Changed

#### Job Logs

You can now download all execution logs for a single provisioning job if it has finished with error. To learn more, see: [Manage Provisioning Job Logs \[page 1600\]](#)

## 23 June 2021 – Identity Provisioning

### Changed

#### skipOperations

The `skipOperations` expression now supports the **update** operation for groups. That means, you can set it in your target systems if you don't want your groups to be updated during the next provisioning job.

To learn more, see: [Transformation Expressions \[page 330\]](#) → **skipOperations**

## 2 June 2021 – Identity Provisioning

### New

#### New systems in all bundles

The following system connectors are now available for all bundle options. You can use them as source and proxy systems:

- [SAP Application Server ABAP](#)
- [SAP S/4HANA On-Premise](#)

To learn more, see: [Provisioning Systems for Bundle Tenants \[page 416\]](#)

---

### Changed

#### SAP S/4HANA On-Premise

A new property, `s4hana.onprem.sap-client`, allows you to set a particular AS ABAP client, which is used as the **sap-client** URL parameter. This way, the URL will redirect to this particular AS ABAP client. To learn more, see:

- [List of Properties \[page 94\]](#) → `s4hana.onprem.sap-client`
  - [SAP S/4HANA On-Premise \(Source\) \[page 625\]](#)
  - [SAP S/4HANA On-Premise \(Target\) \[page 902\]](#)
  - [SAP S/4HANA On-Premise \(Proxy\) \[page 1340\]](#)
- 

## 14 May 2021 – Identity Provisioning

### New

#### Creating destinations for bundle tenants

If you have a bundle tenant, you can now create connectivity destinations in SAP BTP cockpit. This enables you to connect to on-premise systems, such as [SAP Application Server ABAP](#). To do this, you have to enable the new authorization role [Manage Destinations](#).

To learn more, see: [Manage Authorizations \[page 1487\]](#)

---

### New

#### SAP Managed and Customer Managed Systems

If you have purchased **SAP S/4HANA Cloud** (as a bundle solution) and get access to your Identity Provisioning user interface, you may see a specific category, named [SAP Managed](#). It will contain one or more systems of type [SAP S/4HANA Cloud](#), created for you in advance by SAP. Systems in this category communicate via the [SAP\\_COMM\\_1193 communication arrangement](#) and have a predefined version property `s4hana.cloud.api.version=2`.

When you manually create a system, it will appear in another category, named [Customer Managed](#).

To learn more, see: [Customer Managed Systems \[page 88\]](#)

---

## New

### SAP Analytics Cloud – SCIM bulk operations

For [SAP Analytics Cloud](#) target systems, you can now enable SCIM bulk operations for provisioning users. That means, the Identity Provisioning service can write, update, and delete a potentially large collection of users in a single request. To do this, you need to use the following new properties:

- `sac.support.bulk.operation`
- `sac.bulk.operations.max.count`

To learn more, see: [SAP Analytics Cloud \(Target\) \[page 731\]](#)

---

## 21 April 2021 – Identity Provisioning

## Changed

### SAP BTP (Bundle)

If you purchase or already have a global account for SAP BTP, you can get a tenant for the Identity Provisioning service. To create this bundle tenant (free of charge), you have to enable the service from a tile in SAP BTP cockpit.

To learn how to do it, see: [SAP Business Technology Platform Bundle \[page 427\]](#)

---

## Changed

### Local Identity Directory

The [Local Identity Directory](#) systems are now supported for all standalone Identity Provisioning tenants - purchased both **before** and **after** [September 1, 2020](#).

Also, if you have purchased the service **after** [September 1](#), you can now use paging for group members and user's groups, by setting the relevant new properties:

- `idds.group.members.paging.enabled`
- `idds.user.groups.paging.enabled`

To learn more about these properties, see: [List of Properties \[page 94\]](#)

---

## 19 March 2021 – Identity Provisioning

### New

#### SAP Ariba Applications

The Identity Provisioning admin console now supports *SAP Ariba Applications*, which you can use as a source, target, and a proxy system. See:

- [SAP Ariba Applications \(Source\) \[page 496\]](#)
- [SAP Ariba Applications \(Target\) \[page 755\]](#)
- [SAP Ariba Applications \(Proxy\) \[page 1085\]](#)

### Announcement

#### Renamed Systems

With the sunset of the *SAP Cloud Platform* brand, now – **SAP Business Technology Platform** (or **SAP BTP**), the following Identity Provisioning system connectors have been renamed, respectively:

- SAP Cloud Platform Master Data Integration → *SAP Master Data Integration*
- SAP Cloud Platform ABAP Environment → *SAP BTP ABAP environment*
- SAP Cloud Platform Account Members → *SAP BTP Account Members (Neo)*
- SAP Cloud Platform Java/HTML5 Apps → *SAP BTP Java/HTML5 apps (Neo)*
- SAP HANA XS Advanced UAA Server → *SAP BTP XS Advanced UAA (Cloud Foundry)*


## 16 February 2021 – Identity Provisioning

### New

#### SAP SuccessFactors

Identity Provisioning now supports the *sapGlobalUserId* attribute. It is used during provisioning of users from *SAP SuccessFactors* to *Identity Authentication* to correctly generate and synchronize user UUIDs between the two systems.

This attribute is also related to a general error **statusCode 432**, which you might encounter if something goes wrong during the UUID synchronization. To learn more, see:

- [, which you might](#) [Guided Answers: Error statusCode: 432](#) 
- [SAP SuccessFactors \(Source\) \[page 635\]](#)

## 2 February 2021 – Identity Provisioning

### New

#### SAP Cloud for Customer

Apart from target system, you can now use *SAP Cloud for Customer* as a source and a target as well. See:

- [SAP Cloud for Customer \(Source\) \[page 630\]](#)
- [SAP Cloud for Customer \(Target\) \[page 909\]](#)
- [SAP Cloud for Customer \(Proxy\) \[page 1350\]](#)

### New

#### SAP BTP – bundle option

Identity Provisioning is now bundled with *SAP Business Technology Platform* (SAP BTP). You can request an Identity Provisioning bundle tenant (free of charge) by creating an incident. To learn more, see:

- [Obtain a Bundle Tenant \[page 407\]](#)
- [SAP Business Technology Platform Bundle \[page 427\]](#)
- [Provisioning Systems for Bundle Tenants \[page 416\]](#)

## 20 January 2021 – Identity Provisioning

### New

#### SAP Central Business Configuration

The Identity Provisioning admin console now supports *SAP Central Business Configuration*, which you can use as a source, target, and a proxy system. See:

- [SAP Central Business Configuration \(Source\) \[page 536\]](#)
- [SAP Central Business Configuration \(Target\) \[page 804\]](#)
- [SAP Central Business Configuration \(Proxy\) \[page 1173\]](#)

### New

#### SAP Master Data Integration

The Identity Provisioning admin console now supports *SAP Master Data Integration*, which you can use as a source and a proxy system. See:

- [SAP Master Data Integration \(Source\) \[page 609\]](#)
- [SAP Master Data Integration \(Proxy\) \[page 1310\]](#)

## Archived Release Notes

- [Archive - 2020](#)
- [Archive – 2019](#)
- [Archive – 2018](#)
- [Archive – 2017](#)
- [Archive – 2016](#)

### 1.2.1 Release Notes – 2021

Take a look at the Identity Provisioning release notes from the year 2021 on the **What's New** portal: [Identity Provisioning – 2021](#)

### 1.2.2 Release Notes – 2020

Take a look at the Identity Provisioning release notes from the year 2020 on the **What's New** portal: [Identity Provisioning – 2020](#)

### 1.2.3 Release Notes – 2019

Take a look at the Identity Provisioning release notes from the year 2019 on the **What's New** portal: [Identity Provisioning – 2019](#)

### 1.2.4 Release Notes – 2018

## 03 December 2018 – Identity Provisioning

### New

#### Microsoft Active Directory

Besides as a source connector, you can now use *Microsoft Active Directory* also as a target, to provision users. To learn more, see:

- [Microsoft Active Directory \(Source\) \[page 676\]](#)
  - [Microsoft Active Directory \(Target\) \[page 961\]](#)
- 

### New

#### SAP HANA XS Advanced UAA Server

The Identity Provisioning UI now supports *SAP HANA XS Advanced UAA Server*, which you can use as a source, target or proxy system. To learn how, see:

- [SAP HANA XS Advanced UAA Server \(Source\) \[page 519\]](#)
  - [SAP HANA XS Advanced UAA Server \(Target\) \[page 777\]](#)
  - [SAP HANA XS Advanced UAA Server \(Proxy\) \[page 1129\]](#)
- 

### New

#### Real-Time Provisioning

You can now immediately provision newly created users without manually running a job or wait for a scheduled one.

- Currently, this scenario supports only **SAP Cloud Platform Identity Authentication** as a source system.
- You can execute real-time provisioning to **all** target systems.

To learn more, see: [Real-Time Provisioning \[page 1554\]](#)

---

### New

#### Destinations

When you create a new system or edit an existing one, you can now add destinations on both subaccount and subscription level. To learn how, see: [Add a System \[page 1477\]](#)

---



## 07 November 2018 – Identity Provisioning

### New

#### Cloud Foundry UAA Server

Besides as a target connector, you can now use *Cloud Foundry UAA Server* as a source or a proxy, as well. To learn more, see:

- [Cloud Foundry UAA Server \(Source\) \[page 658\]](#)
  - [Cloud Foundry UAA Server \(Target\) \[page 941\]](#)
  - [Cloud Foundry UAA Server \(Proxy\) \[page 1402\]](#)
- 

### New

#### Identity Directory

Besides as a source and a target connector, you can now use *Local Identity Directory* as a proxy system, as well.

To learn more, see: [Configuring Local Identity Directory in Proxy Scenario \[page 1579\]](#)

---

## 23 October 2018 – Identity Provisioning

### New

#### SAP S/4HANA Cloud (Beta)

The Identity Provisioning UI now supports *SAP S/4HANA Cloud*, which you can use as a source, target or proxy system in a beta state. To learn how, see:

- [SAP S/4HANA Cloud \(Source\) \[page 613\]](#)
  - [SAP S/4HANA Cloud \(Target\) \[page 888\]](#)
  - [SAP S/4HANA Cloud \(Proxy\) \[page 1318\]](#)
- 

### Change

#### Identity Directory

Identity Directory is no longer a separate service, nor in beta state. You can now use it as a productive source or target system. Also, you can merge user data from multiple source systems and provision it all in a single *Local Identity Directory* target system. To learn more, see:

- [Configuring Local Identity Directory in Target-Source Scenario \[page 1568\]](#)
  - [Patched and Merged Attributes \[page 1590\]](#)
-

## 17 September 2018 – Identity Provisioning

### New

#### SAP Application Server ABAP

Besides as a source connector, you can now use *SAP Application Server ABAP* as a target or a proxy, as well. To learn more, see:

- [SAP Application Server ABAP \(Source\) \[page 484\]](#)
  - [SAP Application Server ABAP \(Target\) \[page 741\]](#)
  - [SAP Application Server ABAP \(Proxy\) \[page 1059\]](#)
- 

## 1 August 2018 – Identity Provisioning

### New

#### Troubleshooting Guide

The Identity Provisioning now provides a troubleshooting section, which helps you if you encounter some common or specific obstacles during your experience with the service. See: [Monitoring and Troubleshooting \[page 1593\]](#)

---

## 13 July 2018 – Identity Provisioning

### New

#### SAP Business Technology Platform (source, target, proxy)

Besides as a target connector, you can now use *SAP Cloud Platform Java/HTML5 Apps* as a source or a proxy, as well. To learn more, see:

- [SAP Cloud Platform Java/HTML5 \(Source\) \[page 514\]](#)
  - [SAP Cloud Platform Java/HTML5 \(Target\) \[page 774\]](#)
  - [SAP Cloud Platform Java/HTML5 \(Proxy\) \[page 1119\]](#)
-

## 25 May 2018 – Identity Provisioning

### New

#### Audit logs (bundle accounts)

You can now access audit logs to track changes made in your Identity Provisioning account. See: [Access Audit Logs \[page 1607\]](#)

---

### New

#### Bundle scenarios (documentation)

As you know, Identity Provisioning can be consumed either as a standalone service or as part of another product – [SAP Jam Collaboration](#) or [SAP SuccessFactors](#). Now you can learn more about these "bundle" cases. See:

- [Obtain a Bundle Tenant \[page 407\]](#)
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Manage Authorizations \[page 1487\]](#)
- 

## 5 March 2018 – Identity Provisioning

### New

#### Subaccounts

You can enable the Identity Provisioning service on a certain number of subaccounts for your global account. This information is now available in the [Support](#) section of the service user interface.

To learn more, see: [Getting Support \[page 1620\]](#)

---

### Enhancement

#### JSON functions

The [manipulateDate](#) function can now convert [Unix Time Stamp](#) date format (integer number) into standard Java ones (like YYYY-MM-DD). That means, if the source system stores a date as a number of milliseconds, after the transformations this number will be converted and written in the target system as a date in a human readable format.

See: [Transformation Functions \[page 362\]](#) → [manipulateDate](#)

---

## 14 February 2018 – Identity Provisioning

### New

#### Properties

Two new properties help you control the notification e-mails sent when a provisioning job fails:

- `ips.job.notification.ignored.consecutive.failures`
- `ips.job.notification.repeat.on.failure`

Find them on page: [List of Properties \[page 94\]](#)

See also: [Manage Job Notifications \[page 1605\]](#)

## Archived Release Notes

- [2017 \[page 68\]](#)
- [2016 \[page 82\]](#)

## 1.2.5 Release Notes – 2017

Date	Function	Type of Change	Description
2017 – 12 – 28	SAP Analytics Cloud (beta)	New	A new provisioning system is available for both reading and writing entities.  See: <a href="#">SAP Analytics Cloud [page 731]</a>

Date	Function	Type of Change	Description
	Properties	New	<p>Four new properties have been created, as follows:</p> <ul style="list-style-type: none"> <li>SSH properties for reading users and groups in <a href="#">SSH Server (Beta)</a> source systems: <a href="#">ssh.read.groups.command</a> and <a href="#">ssh.read.users.command</a></li> <li>SCIM properties, currently applicable only to <a href="#">SAP Analytics Cloud (Beta)</a> source systems: <a href="#">scim.api.csrf.protection</a> and <a href="#">csrf.token.path</a></li> </ul> <p>See: <a href="#">List of Properties [page 94]</a></p>
	SSH Server (beta)	Enhancement	<p>You can now use the <a href="#">SSH Server (Beta)</a> connector for both reading and writing entities.</p> <p>See: <a href="#">SSH Server (Beta) [page 977]</a></p>
	SAP Hybris Cloud for Customer	Enhancement	<p>SAP Hybris C4C connector has a new API, which requires a new transformation in the Identity Provisioning UI. You can either use the old transformation (which is default), or replace it with the new one, configuring two additional properties.</p> <p>See: <a href="#">SAP Sales Cloud and SAP Service Cloud [page 909]</a></p>
2017 – 11 – 24	Job logs	New	<p>You can now export job execution logs.</p> <p>See: <a href="#">Monitor Provisioning Job Logs [page 1594]</a></p>

Date	Function	Type of Change	Description
2017 – 11 – 09	Properties	New	<p>A new SCIM property, <i>scim.group.members.additional.attributes</i>, allows you to request additional attributes while reading groups from an <b>Identity Authentication</b> source system.</p> <p>Find this property on page: <a href="#">List of Properties [page 94]</a></p>
	Job logs	New	<p>You can set a retention period (7, 14 or 30 days) for your provisioning job logs. By default, your logs are kept for 7 days.</p> <p>See: <a href="#">Monitor Provisioning Job Logs [page 1594]</a></p>
	Identity Authentication (system)	Enhancement	<p>You can now read and write groups in the <b>Identity Authentication</b> system using SCIM API. Previously, you could provision users and groups only through the Identity Authentication UI.</p> <p>See: <a href="#">Identity Authentication [page 702]</a></p>

Date	Function	Type of Change	Description
2017 – 10 – 18	Target systems (beta)	New	<p>The following new target systems (connectors) are available in the Identity Provisioning UI:</p> <ul style="list-style-type: none"> <li>• <a href="#">SSH Server (Beta) [page 977]</a>– It helps you execute bash scripts through SSH connection. The configuration allows you to attach separate scripts per entity life-cycle callback (such as user create, group update, and so on).</li> <li>• <a href="#">SAP HANA Database (Beta) [page 853]</a> – It helps you connect to an SAP HANA Database that is installed on a remote system (cloud or on-premise). You can reach its JDBC SQL port either directly or via an SSH tunnel. Once you access this port, you can provision entities (users and user assignments). You have to configure this target connector according to the location where SAP HANA Database is installed. Cases: <ul style="list-style-type: none"> <li>• <b>Installed on-premise</b> – you need to configure an SSH tunnel and the Cloud Connector control access.</li> <li>• <b>Installed on SAP Business</b></li> </ul> </li> </ul>

Date	Function	Type of Change	Description
			<p><b>Technology Platform (Neo)</b> – you can make a direct connection.</p> <ul style="list-style-type: none"> <li>• <b>Installed on SAP Business Technology Platform (Cloud Foundry)</b> – you have to open an SSH tunnel to a running application container. You also need the <a href="#">Space Developer</a> role, and have to configure a security group that allows the applications in this space to access the JDBC SQL port.</li> </ul>
			<p>→ Remember</p> <p>As these connectors are still in beta state, we recommend that you do not use them in enterprise accounts.</p>
	<b>Job notifications</b>	Enhancement	<p>You can now receive e-mail notifications for successful provisioning jobs that have previously failed.</p> <p>See: <a href="#">Manage Job Notifications [page 1605]</a></p>



Date	Function	Type of Change	Description
2017 – 09 – 25	Identity Directory (beta service)	New	<b>Identity Directory</b> is a beta service in SAP BTP cockpit and depends on the Identity Provisioning service. It provides organizations with a directory for securely storing and managing users and groups in SAP Business Technology Platform.
	Local Identity Directory (system)	New	You can use the <i>Identity Directory</i> as your local source or target system.  See: <a href="#">Configuring Local Identity Directory in Target-Source Scenario [page 1568]</a>
	Value mappings	New	A new JSON expression, <code>valueMapping</code> , allows multiple entity attributes from a source system to be mapped to a single custom attribute in the target. For example, you can take user attributes <i>country</i> + <i>city</i> and map them to a target attribute <i>timezone</i> .  See: <a href="#">Transformation Expressions [page 330]</a> → <b>valueMapping</b>

Date	Function	Type of Change	Description
	Target SCIM systems	Enhancement	<p>As you know, in a target system you can disable (deactivate) entities if they are deleted in the source system, or if there is a condition for them not to be read anymore. For this aim, you need to use the <code>deleteEntity</code> scope in the default target system transformations.</p> <p>Now you can disable such entities in generic SCIM systems which don't support PATCH operations. To do this, use the new system property <code>scim.support.patch.operation</code>, setting it to <code>false</code>.</p> <p>Find this property on page: <a href="#">List of Properties [page 94]</a></p> <p>See also: <a href="#">Transformation Expressions [page 330]</a> → <b>deleteEntity</b></p>
2017 – 09 – 07	SAP Document Center	New	<p>You can now use <b>SAP Document Center</b> as a target system to provision users from other systems. See: <a href="#">SAP Document Center [page 840]</a></p>

Date	Function	Type of Change	Description
2017 – 08 – 10	Properties	New	<p>Use the following new properties to retry entity operations (create, update, delete) that have failed due to timeout or rate limit:</p> <ul style="list-style-type: none"> <li>• <code>ips.failed.request.retry.attempts</code></li> <li>• <code>ips.failed.request.retry.attempts.interval</code></li> </ul> <p>Find these properties on page: <a href="#">List of Properties [page 94]</a></p>
	Target systems	Enhancement	<p><b>Google G Suite</b> and <b>Microsoft Azure AD</b> now support writing both users and groups. See:</p> <ul style="list-style-type: none"> <li>• <a href="#">Google G Suite [page 947]</a></li> <li>• <a href="#">Microsoft Azure Active Directory [page 969]</a></li> </ul>
2017 – 07 – 26	Hybrid scenario	Enhancement	<p>You can now export a created proxy system and then import it as a SCIM repository in SAP Identity Management. See:</p> <ul style="list-style-type: none"> <li>• <a href="#">Hybrid Scenario: SAP Identity Management [page 1565]</a></li> <li>• <a href="#">Export and Import Systems [page 1482]</a></li> </ul>

Date	Function	Type of Change	Description
	Concur system	Enhancement	<p>Concur offers three types of edition sites. The Identity Provisioning service supports the <b>Standard</b> one, which allows you to provision users without grouping them into organization units.</p> <p>If your Concur site requires grouping of users, you need to add some extra JSON code lines into your target transformation. To learn how, see: <a href="#">SAP Concur [page 817]</a></p>
2017 – 07 – 07	Hybrid scenario	New	<p>You can now provision entities from a cloud to an on-premise system (and the other way around) without making a direct connection between them. For this aim, you can use a proxy system. See: <a href="#">Hybrid Scenario: SAP Identity Management [page 1565]</a></p> <div> <p><b>i Note</b></p> <p>Currently, this hybrid scenario is only applicable to SAP Identity Management, used as the on-premise system.</p> </div>

Date	Function	Type of Change	Description
	Source systems	Enhancement	<p><b>Concur</b> and <b>Google G Suite</b>, which you could previously use only as target systems, are now available also as sources. Available operations:</p> <ul style="list-style-type: none"> <li>• <a href="#">Concur</a> supports reading and writing users.</li> <li>• <a href="#">Google G Suite</a> supports reading and writing users, as well as reading groups.</li> </ul>
2017 – 06 – 19	Custom HTTP headers	New	<p>You can pass additional information with the HTTP requests.</p> <p>See: <a href="#">List of Properties [page 94]</a> → <b>ips.http.header.&lt;header_name&gt;</b></p>
2017 – 05 – 31	Skip operations	New	<p>If you want the provisioning job to not execute <i>create</i> or <i>delete</i> operations on entities of a certain type, use the <code>skipOperations</code> scope.</p> <p>See: <a href="#">Transformation Expressions [page 330]</a> → <b>skipOperations</b></p>
	Log personal content	New	<p>Choose whether to enable or disable logging of personal data for provisioned entities.</p> <p>See: <a href="#">List of Properties [page 94]</a> → <b>ips.trace.failed.entity.content</b></p>

Date	Function	Type of Change	Description
2017 – 05 – 05	<b>deleteEntity</b>	New	<p>If an entity is no longer existing or read from the source system, and you want to not delete it but only change its status in the target system, set the <code>deleteEntity</code> scope.</p> <p>See: <a href="#">Transformation Expressions [page 330]</a> → <b>deleteEntity</b></p>
	<b>Job notifications</b>	New	<p>You can now subscribe to receive e-mail notifications about provisioning jobs that finish with error.</p> <p>See: <a href="#">Manage Job Notifications [page 1605]</a></p>

Date	Function	Type of Change	Description
	SCIM properties	Enhancement	<p>You can use the following SCIM properties to search for particular entities:</p> <ul style="list-style-type: none"> <li>• <b>scim.user.filter</b> (<i>source systems</i>) – the service will read only the users matching a set filter expression.</li> <li>• <b>scim.user.unique.attribute</b> (<i>target systems</i>) – if the service tries to recreate an existing user, this property will find the user by a specific attribute, and will only update it.</li> <li>• <b>scim.group.unique.attribute</b> (<i>target systems</i>) – if the service tries to recreate an existing group, this property will find the group by a specific attribute, and will only update it.</li> </ul> <p>See: <a href="#">SCIM System [page 974]</a></p>
2017 – 04 – 03	Source/Target system	New	<p>A new system, <i>Microsoft Azure Active Directory</i>, has been added to the Identity Provisioning user interface. You can use Azure AD as both a source and a target system for provisioning users.</p> <p>See: <a href="#">Microsoft Azure Active Directory [page 969]</a></p>

Date	Function	Type of Change	Description
	<b>Delta read</b>	Enhancement	<p>You can now optimize the amount of data retrieved from SCIM and Identity Authentication source systems, during a provisioning job.</p> <p>See: <a href="#">Manage Full and Delta Read [page 1519]</a></p>
2017 – 02 – 23	<b>Entity deletion</b>	New	<p>For previously existing and provisioned entities, if they have been recently deleted from the source system, you can now decide whether to delete them from the target system or not.</p> <p>See: <a href="#">Manage Deleted Entities [page 1522]</a>.</p>
2017 – 02 – 09	<b>Combo box controls</b>	New	<ul style="list-style-type: none"> <li>• When adding or editing a system, you no longer need to manually enter the destination but you can select it from a combo box.</li> <li>• When adding or editing a target system, you no longer need to manually enter a string of source systems. You can now select the relevant one(s) from a combo box.</li> </ul> <p>See <a href="#">Add a System [page 1477]</a> and <a href="#">Search and Edit a System [page 1480]</a>.</p>



Date	Function	Type of Change	Description
	Delta read	New	<p>You can now optimize the amount of data retrieved from <i>Microsoft AD</i> and <i>SAP SuccessFactors</i> source systems during a provisioning job.</p> <p>See: <a href="#">Manage Full and Delta Read [page 1519]</a></p>
2017 – 01 – 19	Import and export	New	<p>You can now import and export source and target systems.</p> <p>See: <a href="#">Export and Import Systems [page 1482]</a></p>
	Trial use	Announcement	<p>You can now test the trial version of the Identity Provisioning service. To open the user interface, go to the <i>Services</i> section in the SAP BTP cockpit.</p>

## Related Information

[Release Notes – 2016 \[page 82\]](#)

## 1.2.6 Release Notes – 2016

Date	Function	Type of Change	Description
2016 – 12 – 21	User interface	New	<p>You can now access the Identity Provisioning service as a separate HTML5 application. To open the user interface, go to the <a href="#">Services</a> section in SAP BTP cockpit.</p> <p>See: <a href="#">Access Identity Provisioning UI of Standalone Tenants [page 1614]</a></p>
	Source system	New	<p>A new source system, <a href="#">LDAP Server</a>, has been added to the Identity Provisioning user interface.</p> <p>See: <a href="#">LDAP Server [page 669]</a></p>
2016 – 11 – 23	Target system	New	<p>A new target system, <a href="#">CloudFoundry UAA Server</a>, has been added to the Identity Provisioning user interface. You can use this system to write identity and authorization data, such as user accounts and groups.</p> <p>See: <a href="#">Cloud Foundry UAA Server [page 941]</a></p>

Date	Function	Type of Change	Description
	Transformations	Enhancement	<p>Three additional features are now available:</p> <ul style="list-style-type: none"> <li>• <b>ignore</b> - this expression allows you to disable parts of the transformation mapping during provisioning</li> <li>• <b>createEntity</b> - you can set this scope to an entity's attribute to ensure that it is only processed during creation.</li> <li>• <b>randomPassword</b> - a function for generating random passwords, using standard and special characters.</li> </ul> <p>See: <a href="#">Manage Transformations [page 1494]</a></p>
	Entities	Enhancement	<p>You can now provision ABAP roles and transform them as SCIM groups in a target system.</p> <p>See: <a href="#">SAP Application Server ABAP [page 484]</a></p>
2016 – 11 – 09	Source system	New	<p>A new source system, <i>SCIM System</i>, has been added to the Identity Provisioning user interface. You can use this system to provision identity and authorization data.</p> <p>See: <a href="#">SCIM System [page 974]</a></p>
2016 – 10 – 26	Transformations	New	<p>New functions are available for transformations of all source and target systems.</p> <p>See: <a href="#">Manage Transformations [page 1494]</a></p>

Date	Function	Type of Change	Description
	Target systems	New	<p>You can use the following target systems to read provisioned identity data:</p> <ul style="list-style-type: none"> <li>• <a href="#">Google G Suite [page 947]</a></li> <li>• <a href="#">SAP Concur [page 817]</a></li> </ul>
2016 – 10 – 12	Job Execution Details	Enhancement	<p>The function <i>Job Execution Details</i> has now been enhanced to help you investigate any failed entities.</p> <p>See: <a href="#">Monitor Provisioning Job Logs [page 1594]</a></p>
	Target system	New	<p>A new target system, <i>SAP Cloud Platform Java/HTML5 Apps</i>, has been added to the Identity Provisioning user interface. You can use this system to read identity data.</p> <p>See: <a href="#">SAP BTP Java/HTML5 apps (Neo) [page 774]</a></p>
2016 – 09 – 15	Identity Provisioning (service)	New	<p>SAP Cloud Identity Services – Identity Provisioning allows customers to provision the centrally managed identities and their access across the enterprise.</p> <p>See: <a href="#">What Is Identity Provisioning? [page 5]</a></p>

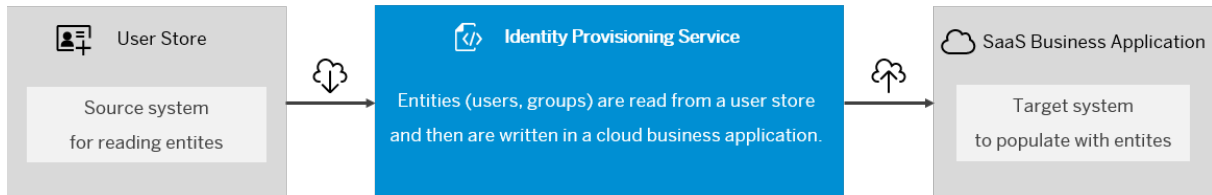
Date	Function	Type of Change	Description
	<b>Identity Provisioning (UI)</b>	New	In SAP Cloud Identity Administration Console, there is a new section – <i>Identity Provisioning</i> . Its purpose is to provide easy provisioning of users, groups and other entities between heterogeneous systems.
	<b>Source systems</b>	New	<p>You can use the following source systems to provision identity and authorization data:</p> <ul style="list-style-type: none"> <li>• <a href="#">SAP Application Server ABAP [page 484]</a></li> <li>• <a href="#">Microsoft Active Directory [page 676]</a></li> <li>• <a href="#">SAP SuccessFactors [page 635]</a></li> <li>• <a href="#">Identity Authentication [page 702]</a></li> </ul>
	<b>Target systems</b>	New	<p>You can use the following target systems to write identity data:</p> <ul style="list-style-type: none"> <li>• <a href="#">Identity Authentication [page 702]</a></li> <li>• <a href="#">SAP Sales Cloud and SAP Service Cloud [page 909]</a></li> <li>• <a href="#">SCIM System [page 974]</a></li> <li>• <a href="#">SAP Jam Collaboration [page 868]</a></li> </ul>

## 1.3 Concepts

This section describes basic concepts that will help you familiarize yourself with the Identity Provisioning service.

The Identity Provisioning service ensures the synchronization of the entities between a source system and one or multiple target systems.

- **Source** – a system, where the company is currently managing the corporate identities
- **Target** – a system that needs to be populated with corporate users and other entities.



You can configure the required provisioning entities in order to ensure proper synchronization between source and target systems. You can also use proxy systems for indirect connections between a system supported by Identity Provisioning and an external application that uses a SCIM 2.0 API to consume identities from the proxy system. For example, you can use SAP Identity Management as an external consuming application.

Properties help you to customize the way your identities are read from a source system or provisioned to the target one. They can also filter which entities and attributes to be read or skipped during the provisioning job.

For every system supported by the Identity Provisioning service, there is an initial (default) transformation logic that converts the system specific JSON representation of the entities from/to one common JSON. You can keep the default transformation, or modify the mapping rules to reflect the current setup of entities from your source or target system.

## Related Information

[System Types \[page 86\]](#)

[Properties \[page 90\]](#)

[Transformations \[page 323\]](#)

### 1.3.1 System Types

Identity Provisioning differentiates systems based on how they are created and what they are used for.

Based on how they are created, provisioning systems are classified as:

- **Customer Managed** - These are the provisioning systems that you as a customer create and configure in the Identity Provisioning UI.
- **SAP Initiated** - These are the provisioning systems that are automatically created and preconfigured in the Identity Provisioning UI. You can modify their initial configuration to make them suitable for your scenario.

Based on their usage, provisioning systems are classified as: **Source**, **Target**, and **Proxy**. All three of them can be created as Customer Managed and SAP Initiated systems.

## Source Systems

A source system is the connector used for reading entities (users, groups, roles). Source systems can be on-premise or cloud-based, SAP or non-SAP, and usually represent the corporate user store where identities

are currently maintained. Identity Provisioning reads the entities from the source system and creates or updates them in the relevant target ones. The provisioning is triggered from the [Jobs](#) tab of a source system.

You can connect one source system to one or multiple target systems.

### → Remember

In the case of multiple (enabled) target systems, when you start a [Read](#) or a [Resync](#) job, this operation will trigger provisioning of entities from this source system to all relevant target ones.

To check the list of all supported source systems, see: [Source Systems \[page 452\]](#)

## Target Systems

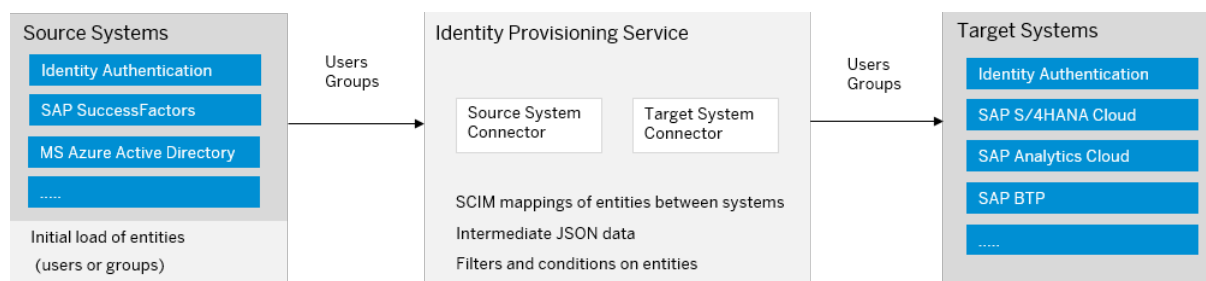
A target system is the connector used for writing (provisioning) entities. Target systems are usually clouds, where Identity Provisioning creates or updates the entities taken from the source system.

A target system can be connected to a single or multiple source systems.

### → Remember

In the case of multiple source systems, we recommend that you run the provisioning jobs **successively** for each system, not simultaneously. This way, you'll avoid incorrect overwriting or merging of entity data, hence failed provisioning jobs.

To check the list of all supported target systems, see: [Target Systems \[page 702\]](#)



## Proxy Systems

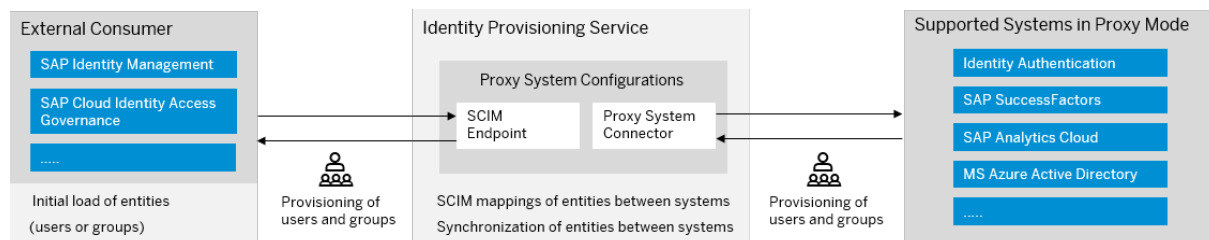
A proxy system is a special connector used for "hybrid" scenarios. It exposes any Identity Provisioning supported backend system as a SCIM 2.0 service provider, which can be consumed by any [SCIM 2.0](#) compatible client application, without making a direct connection between them.

To achieve this, the Identity Provisioning service uses this special proxy system to execute provisioning operations (create, update, delete, etc.) requested by the client application.

The examples in this section cover using of SAP Identity Management as a consuming client application but you can use any other SCIM-based identity management solution. For more information, see: [Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

To provide communication between SAP Identity Management and the back-end system, the proxy application uses a SCIM 2.0 protocol. A system can act as a proxy if it supports both read and write operations.

#### How a proxy system works:



1. The Identity Provisioning service exposes the back end of a supported system as a "proxy".
2. An external application (for example, SAP Identity Management) regards the proxy system as its back-end system.
3. The entities (users) exposed by the back-end system are mapped to SCIM 2.0 entities, if possible. If not possible, the SCIM standard provides a mechanism to define a new resource type with the appropriate schema. You can use the custom resource type to map the back-end entities. See: [SCIM Resources](#) ➔
4. Finally, the external application can start sending REST web service requests to the proxy system in order to read identities from the back end of the SCIM 2.0 system.

To check the list of all supported proxy systems, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Customer Managed Systems \[page 88\]](#)

[System Configuration Details \[page 88\]](#)

[Supported Systems \[page 452\]](#)

### 1.3.1.1 Customer Managed Systems

All provisioning systems (source, target and proxy) that you configure in the Identity Provisioning administration console are displayed under [Customer Managed](#) category.

### 1.3.1.2 System Configuration Details



The system types have similar Identity Provisioning user interface. Below are the details you need to provide when setting up a source, target, or proxy system:

Tab / Field	Description
▮▮ <a href="#">Details</a> ▸ <a href="#">Type</a> ▮	(Mandatory) The type of the source or target system. You can select a particular system from the drop-down list.
▮▮ <a href="#">Details</a> ▸ <a href="#">System Name</a> ▮	(Mandatory) The name of the source or target system configuration. This name will be displayed in the job log and other reports.
▮▮ <a href="#">Details</a> ▸ <a href="#">Destination Name</a> ▮	(Optional) The name of the destination configuration for the system. You define it in the <a href="#">Destinations</a> editor in SAP BTP cockpit. For more information, see <a href="#">Create HTTP Destinations</a> .
	<div><b>i Note</b></div> <p>This field is only mandatory for ABAP systems.</p>
▮▮ <a href="#">Details</a> ▸ <a href="#">Description</a> ▮	(Optional) Enter a meaningful description. It will help you easily distinguish your systems in the list later.
▮▮ <a href="#">Details</a> ▸ <a href="#">Source Systems</a> ▮	<div><b>i Note</b></div> <p>This field is only available for target systems.</p> <p>(Optional) The name or list of names of the source systems that the entities should be read from and transferred to this target system. The list can contain one or more source system names, separated by comma (,).</p> <p>If no source system is specified in this field, the target system receives entities from all source systems configured in the <a href="#">Source Systems</a> tile for the customer tenant.</p>
<a href="#">Transformations</a>	<p>The initial transformation logic is created when saving the source or target system. Every system has specific JSON requirements - these are data models for the entities that have to be synchronized using the Identity Provisioning service. Transformations are settings that represent the logic used to convert or filter the entities data taken from the source before sending it to the target system. Transformations also define how the different attributes of the entities should be mapped. The Identity Provisioning service offers default transformation settings per system, which can be additionally configured. For more information, see: <a href="#">Manage Transformations [page 1494]</a></p>

Tab / Field	Description
<a href="#">Properties</a>	<p>(Optional) You can set properties for the source or target systems. This helps you filtering the data taken from the source system, or to apply a filter to the data before writing it into the target system.</p> <p>These properties overwrite the properties set in the <a href="#">Additional Properties</a> section in <a href="#">SAP BTP cockpit</a> <a href="#">Destinations</a>. For more information, see: <a href="#">Create HTTP Destinations</a></p>
<a href="#">Jobs</a>	<div> <p><b>i Note</b></p> <p>This tab is only available for source systems. It appears once you have successfully configured the source system.</p> </div> <p>From the <a href="#">Jobs</a> tab, you can start or schedule the provisioning job, or resynchronize the data in the target system if changes are made in the source system. For more information, see: <a href="#">Monitor Provisioning Job Logs [page 1594]</a></p>

## 1.3.2 Properties

You need to set mandatory properties to configure the connection between your source and target systems.

For your system provisioning goals, you can set properties in two places:

- SAP BTP cockpit: [Destinations](#)
- Identity Provisioning UI: [Properties](#)

### i Note

If the same properties exist in both the [Destinations](#) editor (in the cockpit) and in the [Properties](#) tab (in the Identity Provisioning UI), the values set in the [Properties](#) tab are taken with higher priority.

Properties help you to customize the way your identities are read from a source system or provisioned to the target one. They can also filter which entities and attributes to be read or skipped during the provisioning job. According to their usability, properties can be categorized as:

- Standard
- Credential
- Default
- Parameterized
- Internal

To learn more, see: [Property Types \[page 91\]](#)

## Related Information

[List of Properties \[page 94\]](#)

[Manage Properties \[page 1505\]](#)

### 1.3.2.1 Property Types

Properties can help you filter which entities and entity attributes are read from the source system or written to the target system. According to their usability, properties can be categorized as follows:

#### Standard System Properties

Each source, target, and proxy system support specific types of properties. For example:

Examples:

AS ABAP System	SAP SuccessFactors
jco.client.r3name=PSE	sf.page.size=100
jco.destination.peak_limit=10	sf.user.filter=firstName John
jco.destination.pool_capacity=5	sf.user.attributes=email

#### Credential Properties

The values of these properties contain sensitive information that must **not** be displayed as plain text. The default credential property name is *Password*, which can represent standard passwords, private keys, or OAuth client secrets. When you add a credential property, its value is displayed as an encrypted string. For better security, the encrypted string is always displayed as 40 characters, no matter how long your password actually is.

##### **i** Note

Properties whose values contain sensitive information can be added only as *Credential* in the Identity Provisioning UI. The values of these properties are stored as encrypted data. They are excluded from the file during system configuration export.

Examples:

SAP HANA Database	SSH Server
hana.jdbc.db.password=*****	ssh.password=*****
hana.jdbc.ssh.tunnel.cf.password=*****	ssh.private.key=*****
hana.jdbc.ssh.tunnel.private.key=*****	ssh.totp.secret.key=*****

### Note

When you update the URL or the host name of an existing provisioning system, you must re-enter the values of the configured credential properties. The only exception to this are the credential properties of systems that are created with a connectivity destination.

## Default System Properties

These properties depend on the particular connector type. They exist in the transformations by default. It is possible to delete some of them but this may cause a loss of provisioned data. Example:

Examples:

LDAP Server
ldap.group.object.class=groupOfNames
ldap.user.object.class=inetOrgPerson
ldap.attribute.user.mobile=mobile
ldap.group.filter=<empty>
ldap.user.filter=<empty>

## Parameterized System Transformations

They use parameters taken from the system property sets. The parameters consist of a *unique key* and a *value*. Like the standard properties, they can be configured in the system's *Properties* tab, and/or in the system's destination properties (in the platform cockpit). When one parameter exists in both property sets, the system properties have priority over the system destination properties. In the JSON data, the unique key of one of these parameters is surrounded by the percent symbol (%). During the transformation evaluation, each occurrence of %<...>% is replaced by the corresponding parameter's value. Parameter references without a value are left unchanged. For example:

Examples:

#### LDAP parameters – list

ldap.attribute.user.mail=mail

ldap.attribute.user.givenName=givenName

ldap.attribute.user.groups=memberOf

#### LDAP parameters – mapping transformation

##### Sample Code

```
/* LDAP Server (source) system: */
{
    "sourcePath":
    "$.%ldap.attribute.user.mail%[0]",
    "targetPath":
    "$.emails[0].value",
    "optional": true
},

{
    "sourcePath":
    "$.%ldap.attribute.user.givenName%[0]",
    "targetPath":
    "$.name.givenName",
    "optional": true
},
```

**NOTE:** Nested parameters are not supported.

## Internal Properties

### Note

Identity Provisioning uses internal properties in various cases. Internal properties must not be used by customers.

For example:

Property Name	Value
destinationName	The name of the destination created in SAP BTP cockpit.

## Related Information

[List of Properties \[page 94\]](#)



[Manage Properties \[page 1505\]](#)

## 1.3.2.2 List of Properties

On this page you can find all the available properties to use in the Identity Provisioning service. You can filter them by system type name, "All Systems", by a word or only part of it.

Name	Description	System Type
Type	<p>Protocol type for making a connection</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"><li><a href="#">HTTP</a></li><li><a href="#">LDAP</a></li><li><a href="#">RFC</a></li></ul> <p><b>System Role:</b> Source, Target, Proxy</p>	All systems
URL	<p>URL needed to make an HTTP(S) connection to an on-premise system or a cloud service</p> <p><b>Possible value:</b></p> <ul style="list-style-type: none"><li>When the proxy type of the HTTP connection is Internet: <b><code>http(s)://&lt;host&gt;:&lt;port&gt;</code></b></li><li>When the proxy type of the HTTP connection is OnPremise: <b><code>http://&lt;virtualhost&gt;:&lt;virtualport&gt;</code></b></li></ul> <p><b>System Role:</b> Source, Target, Proxy</p>	All HTTP systems
ProxyType	<p>Proxy type required for HTTP connection</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"><li><a href="#">Internet</a></li><li><a href="#">OnPremise</a></li></ul> <p><b>System Role:</b> Source, Target, Proxy</p>	All HTTP systems

Name	Description	System Type
Authentication	<p>Authentication type required for HTTP connection</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">NoAuthentication</a></li> <li><a href="#">BasicAuthentication</a></li> <li><a href="#">ClientCertificateAuthentication</a></li> </ul> <p><b>System Role:</b> Source, Target, Proxy</p>	<p>All HTTP systems</p> <div> <p><b>Note</b></p> <p>Identity Provisioning supports certificate-based authentication for secure communication with the provisioning systems (connectors) provided by the service. Refer to the documentation of the respective systems to find out how to upload Identity Provisioning certificates on their end. For example, see <a href="#">How to Create Communication Users</a> in SAP BTP ABAP Environment.</p> </div>
User	<p>It represents:</p> <ul style="list-style-type: none"> <li>User name – used in standard destinations</li> <li>Client ID – used for access token retrieval in OAuth HTTP destinations</li> </ul> <p><b>Possible values:</b> Text/numeric string</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<p>All HTTP systems</p>
Password	<p>(Credential)</p> <p>It represents:</p> <ul style="list-style-type: none"> <li>Password – used in standard destinations</li> <li>Client secret key – used for access token retrieval in OAuth HTTP destinations</li> </ul> <p><b>Possible values:</b> Encrypted string</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<p>All HTTP systems</p>


Name	Description	System Type
<code>abap.user.filter</code>	<p>Filters users by a regular expression on their username. The regex can define any kind of search pattern.</p> <div>  <b>Caution</b>  This property is rather obsolete. For newly created systems, please use <code>abap.user.name.filter</code>. </div> <p><b>Possible values:</b></p> <p>For example: <code>abap.user.filter = ^A.*</code></p> <p>This filter returns all user names that start with capital <code>A</code>.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Application Server ABAP
<code>abap.role.filter</code>	<p>Filters user roles by a regular expression. The regex can define any kind of search pattern.</p> <div>  <b>Caution</b>  This property is rather obsolete. For newly created systems, please use <code>abap.role.name.filter</code>. </div> <p><b>Possible values:</b></p> <p>For example: <code>abap.role.filter = ^order.*</code></p> <p>This filter provisions all roles that start with <code>order</code>.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Application Server ABAP



Name	Description	System Type
<code>abap.user.name.filter</code>	<p>Filters users by a regular expression on their username. The regex can define any kind of search pattern.</p> <p>This property has a higher priority over <code>abap.user.filter</code>. That means, if you set both properties in a system, the value of <code>abap.user.name.filter</code> will be used. However, if the value of <code>abap.user.name.filter</code> is empty, then <code>abap.user.filter</code>'s value will be used instead.</p> <div> <p><b>i Note</b></p> <p>As <code>abap.user.filter</code> is obsolete, we recommend that you use <code>abap.user.name.filter</code>.</p> </div> <p><b>Possible values:</b></p> <p>For example:</p> <pre>abap.user.name.filter = ^MAR.*</pre> <p>This regex reads all user names that start with <i>MAR</i>, such as <i>MARK</i>, <i>MARTINA</i>, and so on.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Application Server ABAP

Name	Description	System Type
<code>abap.role.name.filter</code>	<p>Filters roles by a regular expression. The regex can define any kind of search pattern.</p> <p>This property has a higher priority over <code>abap.role.filter</code>. That means, if you set both properties in a system, the value of <code>abap.role.name.filter</code> will be used. However, if the value of <code>abap.role.name.filter</code> is empty, then <code>abap.role.filter</code>'s value will be used instead.</p> <div> <p><b>i Note</b></p> <p>As <code>abap.role.filter</code> is obsolete, we recommend that you use <code>abap.role.name.filter</code>.</p> </div> <p><b>Possible values:</b></p> <p>For example:</p> <pre>abap.role.name.filter = ^inter.*</pre> <p>This regex reads all roles that start with <i>inter</i>, such as <i>internal</i>, <i>internship</i>, and so on.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Application Server ABAP

Name	Description	System Type
abap.role.prefix	<p>This property distinguishes SAP Application Server ABAP (AS ABAP) roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>ABAP_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the AS ABAP source system and will be provisioned to the target system with the following name pattern: <b>ABAP_&lt;role_name&gt;</b> . This way AS ABAP roles in the target system will be easily distinguished from roles provisioned from other applications. If the property is not set, the AS ABAP roles will be read and provisioned to the target system with their actual role names.</li> <li>When <b>set in the target system</b>, only roles containing the <b>ABAP_</b> prefix in their role name will be provisioned to AS ABAP. Roles without this prefix in their names won't be provisioned. If the property is not set, all roles will be provisioned to AS ABAP.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP Application Server ABAP

Name	Description	System Type
<code>abap.user.membership.filter</code>	<p>Filters users by a regular expression, based on their <a href="#">Role</a> memberships in AS ABAP. The regex can define any kind of search pattern.</p> <p><b>Possible values:</b></p> <p>For example:</p> <pre>abap.user.membership.filter = (?i)^new.*</pre> <p>This reads all users who have an assigned role which starts with <a href="#">new</a>. This regex is case insensitive, which means the result can be roles starting with <a href="#">new</a>, or <a href="#">New</a>, or <a href="#">NEW</a>, and so on.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Application Server ABAP
<code>ariba.applications.api.key</code>	<p>(Credential)</p> <p>This property corresponds to the <a href="#">Application key</a> for your SAP Ariba application. You obtain it during the creation of your application in the SAP Ariba developer portal.</p> <p>See: <a href="#">How to find your application's application key and OAuth client ID</a></p> <p><b>Possible values:</b> Text/numeric string</p> <p>For example:</p> <pre>123abc123XYZ000abc123ABC012345</pre> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Ariba Applications
<code>ariba.applications.realm.id</code>	<p>This property corresponds to the SAP Ariba realm that your application has access to. To learn how to get it, see: <a href="#">How to find your SAP Ariba realm name?</a> </p> <p><b>Possible values:</b> Text/numeric string</p> <p>For example:</p> <pre>123abc123XYZ000abc123ABC012345</pre> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Ariba Applications

Name	Description	System Type
<code>ariba.applications.group.flatten</code>	<p>This property allows or forbids reading "nested groups" (group structures) from SAP Ariba Applications. If enabled (<i>true</i>), group members of type <i>group</i> will be ignored during read in order not to be provisioned to target systems that do not support nested groups.</p> <p><b>Possible values:</b></p> <p>Default value: <i>false</i></p> <p>Predefined value (during system creation):</p> <ul style="list-style-type: none"> <li>• <i>Source systems: true</i> Set it to This property distinguishes SAP Ariba Applications groups by specific prefix. It is an optional property which does not appear by default at system creation. <i>false</i> only if you are sure that the target system supports nested groups.</li> <li>• <i>Proxy systems: false</i> Leave the default/predefinedThis property distinguishes SAP Ariba Applications groups by <i>false</i>Leave the default/predefined value only if you are sure that the consuming external application (identity management system) supports nested groups.</li> </ul> <p><b>System Role:</b> Source, Proxy</p>	SAP Ariba Applications

Name	Description	System Type
ariba.applications.group.prefix	<p>This property distinguishes SAP Ariba Applications groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value:</p> <p><b>ARIBA_APPLICATIONS_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Ariba Applications source system and will be provisioned to the target system with the following name pattern: <b>ARIBA_APPLICATIONS_&lt;GroupDisplayName&gt;</b>. This way SAP Ariba Applications groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Ariba Applications groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>ARIBA_APPLICATIONS_</b> prefix in their display name will be provisioned to SAP Ariba Applications. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Ariba Applications.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP Ariba Applications

Name	Description	System Type
<code>ariba.applications.support.patch.operation</code>	<p>The default value of this property is <i>false</i>. But for SAP Ariba Applications proxy systems, this property appears during creation and its predefined value is <i>true</i>. That means, when the Identity Provisioning identifies a changed entity in the back-end system, it will execute the updates as PATCH requests instead of PUT. That is, only changes will be written in SAP Ariba Applications, instead of provisioning the whole entity data.</p> <p>Note that only attributes without "scope": "createEntity" in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set "scope": "createEntity".</p> <div> <p><b>i Note</b></p> <p>Removing empty groups returns 400 Bad Request instead of 200 OK.</p> </div> <p><b>Possible values:</b></p> <p>Default value: <i>false</i></p> <p>Predefined value (during system creation): <i>true</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Ariba Applications
<code>ariba.applications.user.filter</code>	<p>When specified, only those SAP Ariba Applications users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example: <i>userName eq "SmithJ"</i></p> <p><b>System Role:</b> Source</p>	SAP Ariba Applications

Name	Description	System Type
<code>ariba.applications.group.filter</code>	<p>When specified, only those SAP Ariba Applications groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example: <code>displayName eq "ProjectTeam1"</code></p> <p><b>System Role:</b> Source</p>	SAP Ariba Applications
<code>ariba.applications.content.type</code>	<p>This property makes a SAP Ariba Applications connector to send a specified value for the <code>Content-Type</code> HTTP header. This is needed because SAP Ariba Applications could potentially not implement the protocol in the specification, which states that a system must accept <code>application/scim+json</code> as a value of the <code>Content-Type</code> header.</p> <p><b>Possible values:</b></p> <p>For example: <code>application/json</code></p> <p>Default value: <code>application/scim+json</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Ariba Applications



Name	Description	System Type
<code>ariba.applications.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists in SAP Ariba Applications. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s). This property defines by which unique attribute(s) the existing user to be searched (resolved).</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> <li>• <i>emails[0].value</i></li> <li>• <i>userName,emails[0].value</i></li> </ul> <p>Default value: <i>userName</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Ariba Applications

Name	Description	System Type
<code>ariba.applications.group.unique.attribute</code>	<p>If the Identity Provisioning tries to create a group that already exists on the SAP Ariba Applications target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific). To make the search filter by a specific attribute, specify this attribute as a value for this property.</p> <p><b>Possible values:</b></p> <p>Default value (when not specified): <code>displayName</code></p> <p>If the property is not specified, the search is done by the default attribute: <code>displayName</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Ariba Applications
<code>ariba.applications.include.if.match.wildcard.header</code>	<p>Makes the SAP Ariba Applications connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by an SAP Ariba Applications system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Ariba Applications
<code>c4c.api.version</code>	<p>This property defines the API version that the API of your SAP Sales Cloud and SAP Service Cloud system uses.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>1</code></li> <li><code>2</code></li> <li><code>3</code></li> </ul> <p>By default, Identity Provisioning uses version <code>3</code>, which means - SCIM 2.0 based API.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Sales Cloud and SAP Service Cloud

Name	Description	System Type
c4c.custom.namespace.<prefix>	<div data-bbox="624 376 863 456"> <b>Note</b>  Only relevant to API v.2. </div> <p data-bbox="603 501 986 696">The Identity Provisioning service uses a single predefined namespace for all attributes. However, you can provision entities by defining your own (custom) namespaces for some attributes. For this purpose, you have to:</p> <ol data-bbox="608 719 986 837" style="list-style-type: none"> <li>1. Specify a namespace using this property.</li> <li>2. Set the custom namespace in the JSON transformation.</li> </ol> <p data-bbox="603 860 986 949">For more information, see: <a href="#">SAP Sales Cloud and SAP Service Cloud [page 909]</a></p> <p data-bbox="603 994 767 1016"><b>Possible values:</b></p> <p data-bbox="603 1039 986 1173">The value of this property is the namespace URI. For <b>&lt;prefix&gt;</b>, enter the prefix of the custom XML namespace (for example, <i>a123</i>).</p> <p data-bbox="603 1196 986 1263">Example for setting up the whole property:</p> <p data-bbox="603 1285 986 1375"><i>c4c.custom.namespace.a123=http://sap.com/xi/AP/ CustomerExtension/ABC/A123XX</i></p> <p data-bbox="603 1420 804 1442"><b>System Role:</b> Target</p>	SAP Sales Cloud and SAP Service Cloud

Name	Description	System Type
<code>cbc.user.filter</code>	<p>When specified, only those SAP Central Business Configuration users matching the filter expression will be read.</p> <div> <p><b>i Note</b></p> <p>For source systems only:</p> <p>Using this property makes sense only if you have set the <i>"ignore"</i>: <i>true</i> statement to <i>false</i>.</p> </div> <p><b>Possible values:</b></p> <p>For example: <i>name.familyName eq "Smith" and addresses.country eq "US"</i></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Central Business Configuration
<code>cbc.group.filter</code>	<p>When specified, only those SAP Central Business Configuration groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example: <i>displayName eq "ProjectTeam1" or "Employees2020"</i></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Central Business Configuration

Name	Description	System Type
<code>cbc.group.prefix</code>	<p>This property distinguishes SAP Central Business Configuration groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>CBC_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Central Business Configuration source system and will be provisioned to the target system with the following name pattern: <b>CBC_&lt;GroupDisplayName&gt;</b>. This way SAP Central Business Configuration groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Central Business Configuration groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>CBC_</b> prefix in their display name will be provisioned to SAP Central Business Configuration. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Central Business Configuration.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP Central Business Configuration

Name	Description	System Type
<code>dc.group.filter</code>	<p>This property filters groups by display name or externalId.</p> <p>When specified, only those groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">SAP_Data_Custodian_Auditor</a></li> <li><a href="#">SAP_Data_Custodian_Service_Admin</a></li> <li><a href="#">SAP_Data_Custodian_Key_Admin</a></li> <li><a href="#">SAP_Data_Custodian_Key_User</a></li> </ul> <p>For example: <code>displayName eq "SAP_Data_Custodian_Auditor"</code></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Data Custodian
<code>dc.group.unique.attribute</code>	<p>If the service tries to create a group that already exists in the target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific).</p> <p>To make the search filter by a specific attribute, specify this attribute as a value for the <code>dc.group.unique.attribute</code> property.</p> <p>If the property is not specified, the search is done by the default attribute: <a href="#">displayName</a></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Data Custodian
<code>dc.user.filter</code>	<p>When specified, only those SAP Data Custodian users matching the filter expression will be read. You can filter users by <a href="#">userName</a>, <a href="#">displayName</a> or <a href="#">externalId</a>.</p> <p><b>Possible values:</b></p> <p>For example: <code>userName eq "Smith.J"</code></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Data Custodian



Name	Description	System Type
dc.user.unique.attribute	<p>When Identity Provisioning attempts to provision a user for the first time, it may detect that this user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user will be searched and resolved. If the service finds a user on the target system via this filter, then the conflicting user will overwrite the existing one. If the service does not find a user on the target system via this filter, the creation will fail.</p> <p><b>Default behavior:</b> The property is automatically added during system creation. Its default value is <i>userName</i>. This means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such <i>userName</i> is not found, the creation of the conflicting user fails.</p> <p><b>Possible values:</b></p> <p>Default value: <i>userName</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Data Custodian
dc.content.type	<p>Makes the connector send a specified value for the <i>Content-Type</i> HTTP header. This is needed because a SCIM system could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>If the property is not specified, the default value is taken: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Data Custodian

Name	Description	System Type
dc.include.if.match.wildcard.header	<p>Makes the SAP Data Custodian connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by an SAP Data Custodian system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Data Custodian




Name	Description	System Type
<code>dc.support.patch.operation</code>	<p>This property controls how modified users in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, Identity Provisioning sends a PATCH request to the user or group resource in the target system. Only attributes without "scope": "createEntity" in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set "scope": "createEntity".</li> <li>If set to <i>false</i>, PUT operations are used to update users in the target system. This means, for example, that if a user attribute is modified, all user attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p>Users can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>In the source system, some user attributes are modified, or new attributes are added.</li> <li>In the source system, a condition or a filter is set for users not to be read anymore.</li> <li>A user is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <b>deleteEntity</b>.</p>	SAP Data Custodian

Name	Description	System Type
	<p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	
dc.group.prefix	<p>This property distinguishes SAP Data Custodian groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>DC_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Data Custodian source system and will be provisioned to the target system with the following name pattern: <b>DC_&lt;GroupDisplayName&gt;</b>. This way SAP Data Custodian groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Data Custodian groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>DC_</b> prefix in their display name will be provisioned to SAP Data Custodian. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Data Custodian.</li> </ul> <p><b>System Role:</b> Source, Target</p>	SAP Data Custodian

Name	Description	System Type
<code>hcp.application.names</code>	<p>Enter a comma-separated list of application names. That could be applications deployed on your account, or applications for which your account has subscribed. The property returns the roles assigned to these applications.</p> <p><b>Possible values:</b></p> <p>Use the following format (no spaces):</p> <p><i>&lt;app_name1&gt;,&lt;app_name2&gt;,&lt;provider_subaccount&gt;:&lt;provider_app&gt;</i></p> <p>For example:</p> <p><i>myapp1,myapp2,provider1:app123,provider2:cloud789,mynewapp</i></p> <div>  <b>Caution</b>            You must not leave this property with an empty value.         </div> <p><b>System Role:</b> Source</p>	SAP BTP Java/HTML5 apps (Neo)
<code>hcp.read.group.roles</code>	<p>If you set this property to <i>true</i>, the Identity Provisioning will read the following additional attributes for a SAP Business Technology Platform group:</p> <ul style="list-style-type: none"> <li>• Application roles</li> <li>• Group mappings, defined by your identity provider</li> </ul> <div>  <b>Restriction</b>            This property is applicable only if SAP Business Technology Platform and the external SCIM-based system belong to one and the same region.         </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Proxy</p>	SAP BTP Java/HTML5 apps (Neo)

Name	Description	System Type
hcp.group.prefix	<p>This property distinguishes groups from Java/HTML5 applications running on SAP BTP, Neo environment by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>HCP_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Business Technology Platform source system and will be provisioned to the target system with the following name pattern: <b>HCP_&lt;GroupDisplayName&gt;</b>. This way groups from Java/HTML5 applications running on SAP BTP, Neo environment in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the groups from Java/HTML5 applications running on SAP BTP, Neo environment will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>HCP_</b> prefix in their display name will be provisioned to SAP Business Technology Platform. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Business Technology Platform.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP BTP Java/HTML5 apps (Neo)

Name	Description	System Type
<code>hcp.patch.response.with.re source</code>	<p>Use this property when you execute hybrid scenarios with SAP Business Technology Platform (Neo) as a SCIM proxy system, and you update an entity (mostly relevant to groups, like when you change the members of a group) via a PATCH request.</p> <p>If you set this property to <i>true</i>, the successful PATCH request will return a response code 200 (<i>OK</i>) back to the consumer client application with a payload body containing the updated attributes of the relevant group.</p> <p>If you don't specify the property (or it's set to <i>false</i>), the successful PATCH request will return a response code 204 (<i>No Content</i>) indicating successful group update but with no payload body.</p> <p>For more information, see: <a href="#">SCIM 2.0: Modifying with PATCH</a> .</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Proxy</p>	SAP BTP Java/HTML5 apps (Neo)

Name	Description	System Type
ids.group.filter	<p>This property filters groups by display name. You can set a single display name or multiple ones as filter criteria. If you enter multiple display names (using OR operator), the filter will search for any of them.</p> <p>Value pattern (single): <i>displayName eq "&lt;group_name&gt;"</i></p> <p>Value pattern (multiple): <i>displayName eq "&lt;group_name1&gt;" or displayName eq "&lt;group_name2&gt;"</i></p> <p><b>Possible values:</b></p> <p>For example:</p> <ul style="list-style-type: none"> <li>Single: <i>displayName eq "FellowshipTeam1"</i></li> <li>Multiple: <i>displayName eq "FellowshipTeam1" or displayName eq "JuniorTest3"</i></li> </ul> <p><b>System Role:</b> Source, Proxy</p>	Local Identity Directory

Name	Description	System Type
<code>idds.user.filter</code>	<p>This property filters users by particular attributes. You can set a single attribute or multiple ones as search criteria.</p> <p>Value pattern (single): <code>&lt;user_attribute&gt; eq "&lt;value&gt;"</code></p> <p>Value pattern (multiple):  <code>&lt;user_attribute1&gt; eq "&lt;value1&gt;" and/or &lt;user_attribute2&gt; eq "&lt;value2&gt;"</code></p> <p><b>Possible values:</b></p> <p>For example:</p> <ul style="list-style-type: none"> <li>Single: <code>userName eq "Sebastian"</code></li> <li>Multiple (with OR): <code>userName eq "Sebastian" or addresses.country eq "France"</code></li> <li>Multiple (with AND): <code>userName eq "Sebastian" and addresses.country eq "France"</code></li> <li>Multiple (with brackets):  <code>userName eq "Sebastian" or (addresses.country eq "France" and emails.value eq "sebastian123@mail.com")</code></li> <li>Multiple (enterprise attributes):  <code>urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:department eq "Dev" and urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:organization eq "Technology"</code></li> </ul> <p><b>System Role:</b> Source, Proxy</p>	Local Identity Directory
<code>idds.group.members.paging.enabled</code>	<p>This property enables paging of group members.</p> <p>The maximum number of group members returned per request is 20 000. To read more than 20 000 group members, paging must be enabled.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> - Paging is enabled.</li> <li><code>false</code> - Paging is disabled.</li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Source, Proxy</p>	Local Identity Directory

Name	Description	System Type
<code>idds.user.groups.paging.enabled</code>	<p>This property enables paging of user's groups.</p> <p>The maximum number of user's groups returned per request is 1000. To read more than 1000 user's groups, paging must be enabled.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> - Paging is enabled.</li> <li><code>false</code> - Paging is disabled.</li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Source, Proxy</p>	Local Identity Directory
<code>ips.date.variable.format</code>	<p>This is a default property that the Identity Provisioning UI automatically adds to the configuration of every newly created system. The property allows you to change the default date format to another, more suitable for your scenario.</p> <p>See also: <a href="#">Transformation Variables [page 399]</a>.</p> <p><b>Possible values:</b></p> <p>Default value:</p> <p><code>yyyy-MM-dd HH:mm:ss.SSS</code></p> <p><b>System Role:</b> Source, Target, Proxy</p>	All systems



Name	Description	System Type
<code>ips.delete.existedbefore.entities</code>	<p><b>Use case:</b> An entity exists on the target system, and then a provisioning job reads the same entity from a source system and updates it on the target. If later on you delete this entity from the source system, the next provisioning job will recognize it as a "previously existed one" and will <b>not delete</b> it from the target.</p> <p>If you want such <i>recognized</i> entities to be deleted from the target as well, open the relevant target system and set this property to <b>true</b>.</p> <p>For more information, see <a href="#">Manage Deleted Entities [page 1522]</a>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	All systems

Name	Description	System Type
<code>ips.override.existedbefore.assignments</code>	<p>This property defines whether or not the Identity Provisioning service to overwrite user/group assignments that have existed in the target system before you start provisioning entities to that system.</p> <p><b>Example:</b> Let's say there is a group in the target system that contains some assignments (users and subgroups). In the source system there is a matching group, which contains different assignments.</p> <ul style="list-style-type: none"> <li>• If you start a provisioning job without setting this property (by default, it's <i>true</i>), all assignments from the source group will overwrite the ones from the target group.</li> <li>• If you set the property to <i>false</i>, all existing assignments will be kept in the target system group, and the new ones will just be added.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>true</i></p> <p><b>System Role:</b> Target</p>	SAP Application Server ABAP

Name	Description	System Type
<code>ips.failed.request.retry.attempts</code>	<p>If an entity operation fails due to an occurred exception (rate limit, bad gateway, missing authorization, or timeout), you can specify a number of retries for this operation. Use this property to set the number of retries.</p> <div> <p><b>i Note</b></p> <p>Rate limit is the controlled rate of requests sent to a system. Some systems implement rate limit to avoid overloading and performance issues.</p> </div> <p><b>Possible values:</b></p> <p>Default value: <a href="#">2</a></p> <p>Maximum value: <a href="#">3</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	<p>All systems</p> <div> <p><b>i Note</b></p> <p>For more information about the cases in which the retry is supported, see <a href="#">Handle Failed Operations [page 1533]</a>.</p> </div>
<code>ips.failed.request.retry.attempts.interval</code>	<p>Specify a time interval (in seconds) between the retries, in case an operation fails due to an occurred exception.</p> <p>This property is related to <code>ips.failed.request.retry.attempts</code>.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">30</a></p> <p>Maximum value: <a href="#">60</a></p> <div> <p><b>i Note</b></p> <p>Following an HTTP 502 Bad Gateway server error, the time interval for SAP BTP XS Advanced UAA (Cloud Foundry) and SAP Sales Cloud and SAP Service Cloud must not exceed 50 (seconds).</p> </div> <p><b>System Role:</b> Source, Target, Proxy</p>	<p>All systems</p> <div> <p><b>i Note</b></p> <p>For more information about the cases in which the retry is supported, see <a href="#">Handle Failed Operations [page 1533]</a>.</p> </div>

Name	Description	System Type
<code>ips.job.notification.repeat.on.failure</code>	<p>If you have activated notifications for a source system and a provisioning job fails, you'll receive notification e-mails with subject <i>Provisioning Finished with Error</i>. You can also receive an e-mail if you manually stop a running job.</p> <p>With property <code>ips.job.notification.repeat.on.failure</code>, you can control the frequency of the received notifications.</p> <ul style="list-style-type: none"> <li>• If you set the property to <i>true</i>, you will receive notification e-mails every time a job fails.</li> <li>• If you want to stop or control the notification frequency, set the property to <i>false</i> (default value).</li> </ul> <p>This property has a higher priority than <code>ips.job.notification.ignored.consecutive.failures</code>.</p> <p>See also: <a href="#">Manage Job Notifications [page 1605]</a>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i>. That means, when a job fails, only one notification e-mail will be sent.</p> <p><b>System Role:</b> Source</p>	All systems

Name	Description	System Type
<code>ips.job.notification.ignored.consecutive.failures</code>	<p>If you have activated notifications for a source system and a provisioning job fails, you'll receive notification e-mails with subject <i>Provisioning Finished with Error</i>. You can also receive an e-mail if you manually stop a running job.</p> <p>With property <code>ips.job.notification.ignored.consecutive.failures</code>, you can control the number of the received consecutive notifications.</p> <div> <p><b>Note</b></p> <p>Property <code>ips.job.notification.repeat.on.failure</code> must be set to <i>false</i> or not specified at all.</p> </div> <p><b>Example:</b> If you set <code>ips.job.notification.ignored.consecutive.failures = 3</code> and the job is constantly failing, the first three times you'll not receive a notification. On the fourth job fail, you will receive one notification e-mail. No subsequent e-mails will be sent by the service until the first successful run of the job.</p> <p>See also: <a href="#">Manage Job Notifications [page 1605]</a>.</p> <p><b>Possible values:</b></p> <p>Default value: <i>0</i>.</p> <p>That means, a notification e-mail will be sent after the first job fail.</p> <p><b>System Role:</b> Source</p>	All systems

Name	Description	System Type
<code>ips.job.notification.skip.intermediate.notifications</code>	<p>If you have activated notifications for a source system, and an entity fails during the provisioning job, you'll receive one notification e-mail with subject <i>Provisioning Running with Error</i>.</p> <p>Property</p> <p><code>ips.job.notification.skip.intermediate.notifications</code> controls whether you will receive a notification or not.</p> <ul style="list-style-type: none"> <li>• If the property is set to <i>true</i>, no notifications will be sent.</li> <li>• If the property is not specified or is set to <i>false</i> (default), you'll receive one notification e-mail. No subsequent e-mails will be sent by the service until the first successful run of the job.</li> </ul> <p>See also: <a href="#">Manage Job Notifications [page 1605]</a>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i>. That means, after the first failed entity, a notification e-mail will be sent.</p> <p><b>System Role:</b> Source</p>	All systems

Name	Description	System Type
<code>ips.trace.failed.entity.content</code>	<p>If a provisioning job repeatedly fails and you need problem investigation, you can enable logging and tracing for the personal and sensitive data of your provisioned entities. To do this, set this property to <i>true</i>.</p> <p>If the property is not set, in the logs you see: <code>content = &lt;hidden content&gt;</code></p> <p>To learn more about personal and sensitive data, see:</p> <ul style="list-style-type: none"> <li>• <a href="#">Glossary for Data Protection and Privacy [page 1549]</a></li> <li>• <a href="#">Customer Data [page 1545]</a> → <b>Data Storage Security</b></li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Source</p>	All systems

Name	Description	System Type
<code>ips.trace.skipped.entity.content</code>	<p>If a provisioning job results in skipping entities from source or target systems, you can view the details for each skipped user and group.</p> <p>To do this, you need to enable logging and tracing for the personal and sensitive data of your provisioned entities by setting the property to <i>true</i>.</p> <p>If the property is not set, in the logs you see: <code>content = &lt;hidden content&gt;</code></p> <p>To learn more about personal and sensitive data, see:</p> <ul style="list-style-type: none"> <li>• <a href="#">Glossary for Data Protection and Privacy [page 1549]</a></li> <li>• <a href="#">Customer Data [page 1545]</a> → <b>Data Storage Security</b></li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Source</p>	All systems



Name	Description	System Type
<code>ips.trace.skipped.entity</code>	<p>This property allows you to download and view the details of all skipped entities for a given job in a zip archive. For more information, see <a href="#">Manage Provisioning Job Logs [page 1600]</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> - The downloaded zip file contains all skipped entities for the given job, the systems they are skipped from, the reason behind this, as well as the content of the entities (if <code>ips.trace.skipped.entity.content</code> is set to <code>true</code>).</li> <li><code>false</code> - The downloaded zip file is empty.</li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Source</p>	All systems
<code>ips.http.header.&lt;header_name&gt;</code>	<p>Use this property to pass additional information with the HTTP requests.</p> <p>The provisioning system may override your custom HTTP headers, if specific header settings are implemented in the system.</p> <p><b>Possible values:</b></p> <p>Example for an authorization header:</p> <pre>ips.http.header.authorization = Basic VDAwdfhjgHGSzmfNA==</pre> <div> <p><b>Note</b></p> <p>If you provide credentials for the provisioning system, this property will not take effect. Its value (token) will be overridden by the token generated by the system implementation.</p> </div> <p><b>System Role:</b> Source, Target, Proxy</p>	All HTTP systems

Name	Description	System Type
<code>ips.delta.read</code>	<p>If this property is enabled, every time a provisioning job is started, it does not retrieve the entire amount of source system data but only the last changed entities.</p> <p>For more information, see <a href="#">Manage Full and Delta Read [page 1519]</a>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>enabled</code></li> <li><code>disabled</code></li> </ul> <p><b>System Role:</b> Source</p>	<ul style="list-style-type: none"> <li>Identity Authentication</li> <li>Local Identity Directory</li> <li>Microsoft Active Directory</li> <li>SAP Data Custodian</li> <li>SAP SuccessFactors</li> <li>SAP SuccessFactors Learning</li> <li>SCIM System</li> <li>SAP Central Business Configuration</li> </ul>
<code>ips.full.read.force.count</code>	<p>If your system (connector) works in <a href="#">delta read</a> mode, it's recommended to enforce full reads from time to time. To achieve this, set this property to an integer number.</p> <p><b>Possible values:</b></p> <p>For example: <code>10</code></p> <p>This value results in alternating full reads after every 10 delta reads are performed.</p> <p>In case the property is not set, only delta read jobs will be executed. For more information, see <a href="#">Manage Full and Delta Read [page 1519]</a>.</p> <p><b>System Role:</b> Source</p>	<p>All, except for:</p> <ul style="list-style-type: none"> <li>SAP Application Server ABAP</li> <li>SSH Server (Beta)</li> </ul>

Name	Description	System Type
OAuth2TokenServiceURL	<p>If you need to make OAuth authentication to the system, enter the URL to the access token provider service.</p> <p><b>Possible values:</b> Access token URL</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>• Cloud Foundry UAA Server</li> <li>• Google G Suite</li> <li>• Microsoft Azure Active Directory</li> <li>• Sales Cloud – Analytics &amp; AI</li> <li>• SAP Analytics Cloud</li> <li>• SAP Ariba Applications</li> <li>• SAP BTP Account Members (Neo)</li> <li>• SAP BTP Java/HTML5 apps (Neo)</li> <li>• SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>• SAP Build Work Zone, advanced edition</li> <li>• SAP Build Work Zone, standard edition</li> <li>• SAP Business Network</li> <li>• SAP Central Business Configuration</li> <li>• SAP Commerce Cloud</li> <li>• SAP Data Custodian</li> <li>• SAP Fieldglass</li> <li>• SAP Jam Collaboration</li> <li>• SAP Master Data Integration</li> <li>• SAP Advanced Financial Closing</li> <li>• SAP S/4HANA for procurement planning</li> <li>• SAP Advanced Financial Closing</li> <li>• SCIM System</li> </ul>

Name	Description	System Type
OAuth2TokenScope	<p>If your backend system is OAuth protected and requires an access token with scope, use this property to specify the scope. It defines the token's level of access to protected resources.</p> <p>This is an optional property. Providing a value only makes sense if the backend system and its trusted OAuth server requires it. For more information, see <a href="#">OAuth Scopes</a> .</p> <p>Example value of Google G Suite OAuth scope: <a href="https://www.googleapis.com/auth/admin.directory.user">https://www.googleapis.com/auth/admin.directory.user</a></p> <p>Although Google G Suite OAuth scopes look like URLs, they are not web pages. They are access token permissions.</p> <p><b>Possible values:</b> Scopes are system specific.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>• Cloud Foundry UAA Server</li> <li>• Google G Suite</li> <li>• Microsoft Azure Active Directory</li> <li>• Sales Cloud – Analytics &amp; AI</li> <li>• SAP Analytics Cloud</li> <li>• SAP Ariba Applications</li> <li>• SAP BTP Account Members (Neo)</li> <li>• SAP BTP Java/HTML5 apps (Neo)</li> <li>• SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>• SAP Build Work Zone, advanced edition</li> <li>• SAP Build Work Zone, standard edition</li> <li>• SAP Business Network</li> <li>• SAP Central Business Configuration</li> <li>• SAP Commerce Cloud</li> <li>• SAP Data Custodian</li> <li>• SAP Fieldglass</li> <li>• SAP Jam Collaboration</li> <li>• SAP Master Data Integration</li> <li>• SAP Advanced Financial Closing</li> <li>• SAP S/4HANA for procurement planning</li> <li>• SCIM System</li> </ul>
jco.client.user	<p>Enter the user for AS ABAP.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Application Server ABAP
jco.client.passwd	<p>(Credential)</p> <p>Enter the password for the AS ABAP user.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Application Server ABAP
jco.client.ashost	<p>Enter the virtual host entry that you have configured in the Cloud connector → <a href="#">Access Control</a> configuration.</p> <p><b>Possible values:</b></p> <p>For example: <a href="#">abapserver.hana.cloud</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Application Server ABAP

Name	Description	System Type
<code>jco.client.client</code>	<p>Enter the client to be used in the ABAP system. Valid format is a three-digit number.</p> <p><b>Possible values:</b></p> <p>For example: <a href="#">001</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Application Server ABAP
<code>jco.client.r3name</code>	<p>Enter the three-character system ID of the ABAP system to be addressed.</p> <p><b>Possible values:</b></p> <p>For example: <a href="#">WPE</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Application Server ABAP
<code>jco.client.sysnr</code>	<p>Enter the "system number" of the ABAP system.</p> <p><b>Possible values:</b></p> <p>For example: <a href="#">42</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Application Server ABAP
<code>jco.destination.proxy_type</code>	<p>Defines the proxy type of the connection you need to provide for your ABAP system.</p> <p>The proxy type <a href="#">OnPremise</a> requires the Cloud Connector to access resources within your on-premise network.</p> <p><b>Possible values:</b> <a href="#">OnPremise</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Application Server ABAP
<code>jco.destination.peak_limit</code>	<p>Represents the maximum number of active connections that can simultaneously be created for a destination.</p> <p><b>Possible values:</b></p> <p>For example: <a href="#">10</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Application Server ABAP

Name	Description	System Type
<code>jco.destination.pool_capacity</code>	Represents the maximum number of idle connections kept open by the destination.  <b>Possible values:</b>  For example: <a href="#">5</a>  <b>System Role:</b> Source, Target, Proxy	SAP Application Server ABAP
<code>jco.client.mshost</code>	Represents the message server host to be used.  <b>System Role:</b> Source, Target, Proxy	SAP Application Server ABAP
<code>X-ConsumerKey</code>	(Credential)  Enter the Concur access token needed for the connection.  <b>System Role:</b> Target, Proxy	SAP Concur
<code>jwt.subject</code>	Enter the Google G Suite user on behalf of which the Google Directory API is called.  <b>System Role:</b> Target, Proxy	Google G Suite
<code>jwt.scope</code>	Enter space-separated Google Directory API authorization scopes.  <b>System Role:</b> Target, Proxy	Google G Suite
<code>ldap.url</code>	URL needed to make an LDAP connection to an on-premise system or a cloud service  <b>Possible values:</b> <code>ldap://&lt;host&gt;&lt;port&gt;</code>  <b>System Role:</b> Source, Target, Proxy	<ul style="list-style-type: none"> <li>• LDAP Server</li> <li>• Microsoft Active Directory</li> </ul>
<code>ldap.proxyType</code>	Proxy type for the LDAP connection  <b>Possible values:</b> <a href="#">OnPremise</a>  <b>System Role:</b> Source, Target, Proxy	<ul style="list-style-type: none"> <li>• LDAP Server</li> <li>• Microsoft Active Directory</li> </ul>
<code>ldap.authentication</code>	Authentication type for the LDAP connection  <b>Possible values:</b> <a href="#">BasicAuthentication</a>  <b>System Role:</b> Source, Target, Proxy	<ul style="list-style-type: none"> <li>• LDAP Server</li> <li>• Microsoft Active Directory</li> </ul>

Name	Description	System Type
<code>ldap.user</code>	<p>User name for the LDAP Server</p> <p><b>Possible values:</b> Text/numeric string</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<code>ldap.password</code>	<p>(Credential) Password for the LDAP Server user</p> <p><b>Possible values:</b> Encrypted string</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<code>ldap.group.path</code>	<p>Enter the complete path to the node containing the groups in the LDAP tree.</p> <div> <p>→ Remember</p> <p>We strongly recommend that you enter different paths for LDAP users and groups. That means, the value of <code>ldap.group.path</code> should be different than the value of <code>ldap.user.path</code>.</p> </div> <p><b>Possible values:</b></p> <p>For example:</p> <p><code>ldap.group.path=OU=Groups,OU=IAS,DC=global,DC=corp,DC=mycompany</code></p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<code>ldap.user.path</code>	<p>Enter the complete path to the users in the LDAP Server or Microsoft AD.</p> <div> <p>→ Remember</p> <p>We strongly recommend that you enter different paths for LDAP users and groups. That means, the value of <code>ldap.users.path</code> should be different than the value of <code>ldap.group.path</code>.</p> </div> <p><b>Possible values:</b></p> <p>For example:</p> <p><code>ldap.user.path=OU=Users,OU=IAS,DC=global,DC=corp,DC=mycompany</code></p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>

Name	Description	System Type
<code>ldap.user.attributes</code>	<p>Shows which user attributes to be read from the LDAP server (and respectively, from the intermediate JSON data). Separate the attributes by comma (,).</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>LDAP server version 1 - Even though you specify which user attributes you want to read from the LDAP server, all user attributes are read.</li> <li>LDAP server version 2 - Only the specified attributes will be read from the LDAP server.</li> </ul> <p>If nothing is set, all attributes are read.</p> <p><b>System Role:</b> Source, Proxy</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<code>ldap.group.attributes</code>	<p>Shows which group attributes to be read from the LDAP server (and respectively, from the intermediate JSON data). Separate the attributes by comma (,).</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>LDAP server version 1 - Even though you specify which group attributes you want to read from the LDAP server, all user and group attributes are read.</li> <li>LDAP server version 2 - Only the specified attributes will be read from the LDAP server.</li> </ul> <p>If nothing is set, all attributes are read.</p> <p><b>System Role:</b> Source, Proxy</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>



Name	Description	System Type
<code>ldap.user.object.class</code>	<p>Criteria for user. In the intermediate JSON data, the following LDAP filter is used: (objectClass=user)</p> <p>For target LDAP systems: this property defines the set of supported and required attributes for an LDAP user entity.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">inetOrgPerson</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>• LDAP Server</li> <li>• Microsoft Active Directory</li> </ul>
<code>ldap.group.object.class</code>	<p>Criteria for group. In the intermediate JSON data the following LDAP filter is used: (objectClass=group)</p> <p>For target LDAP systems: this property defines the set of supported and required attributes for an LDAP group entity.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">groupOfNames</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>• LDAP Server</li> <li>• Microsoft Active Directory</li> </ul>

Name	Description	System Type
<code>ldap.group.uniquename.attribute</code>	<p>By default, the <i>memberOf</i> array in the source JSON data contains the CN part of the complete distinguished name of the groups to which the entity belongs.</p> <p>An administrator can use this property to change the default behavior and specify an attribute name to be used instead of CN.</p> <div> <p><b>i Note</b></p> <ul style="list-style-type: none"> <li>Any group that doesn't have the attribute specified, will not be part of the resulting <i>memberOf</i> JSON array.</li> <li>Any group that doesn't match the <code>ldap.group.path</code> property, will not be part of the resulting <i>memberOf</i> JSON array.</li> </ul> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>cn</i> (default for LDAP)</li> <li><i>displayName</i> – this will produce a <i>memberOf</i> array which contains the <i>displayName</i> attribute value of the groups to which the entity belongs.</li> </ul> <div> <p><b>i Note</b></p> <p>Whatever value you choose for this property, it should correspond to the one in the JSON transformation of your LDAP system (the <i>group</i> mapping).</p> </div> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>

Name	Description	System Type
<code>ldap.attribute.group.id</code>	<p>This property denotes the ID of a group.</p> <ul style="list-style-type: none"> <li>When a user is a member of a group, this group is returned in the <i>memberOf</i> array for this user. This property evaluates the attribute used as ID of this group.</li> <li>When a group is a member of another group, this property evaluates the attribute used as ID of the "parent group". In this case, the <code>ldap.attribute.group.id</code> property has a higher priority than <code>ldap.group.uniqueName.attribute</code>.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>cn</i> (default)</li> <li><i>distinguishedName</i> – this will produce a <i>memberOf</i> array which contains the <i>distinguishedName</i> attribute value of the groups to which the entity belongs.</li> </ul> <div> <p><b>Note</b></p> <p>Whatever value you choose for this property, it should correspond to the one in the JSON transformation of your LDAP system (the <i>group</i> mapping).</p> </div>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<b>System Role:</b> Source, Target, Proxy		


Name	Description	System Type
<code>ldap.attribute.dn</code>	<p>This property denotes the distinguished name of a user or a group.</p> <p>The distinguished name is automatically assigned and cannot be configured.</p> <p>The behavior described below is valid only when Microsoft Active Directory is used as target system:</p> <p>When the Identity Provisioning attempts to provision a user or a group to Microsoft Active Directory for the first time, it may detect that such a user or group already exists on the target system. Thus, the service needs to retrieve the entityId of the existing user or group by using this property for conflict resolution.</p> <p>If the service finds such a user on the target system via this filter, the creation will fail. In this case, the conflicting user will overwrite the existing one.</p> <p>If the service finds such a group on the target system via this filter, the creation will fail. In this case, the existing group only needs to be updated.</p> <p>If the service does not find a user or a group via this filter, the creation will fail.</p> <p><b>Possible values:</b></p> <p>Default and only possible value:  <a href="#"><i>distinguishedName</i></a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>• LDAP Server</li> <li>• Microsoft Active Directory</li> </ul>

Name	Description	System Type
<code>ldap.member.uniquename.attribute</code>	<p>Determines the value of the member attribute of groups in the intermediate JSON data.</p> <p>This property can return either the common name (CN) of the user or the entire distinguished name (DN).</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>cn</code> (default for Microsoft AD)</li> <li><code>uid</code> (default for LDAP Server)</li> <li><code>distinguishedName</code></li> </ul> <div> <p><b>Note</b></p> <p>Whatever value you choose for this property, it should correspond to the one in the JSON transformation of your system (the <code>user</code> mapping).</p> </div>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<code>ldap.attribute.user.id</code>	<p>This property denotes the ID of a user.</p> <p>When a user is a member of a group, this property evaluates the attribute used as ID of this member. In this case, the <code>ldap.attribute.user.id</code> or <code>ldap.attribute.group.id</code> property has a higher priority than <code>ldap.member.uniquename.attribute</code>.</p> <p><b>Possible values:</b></p> <p>Possible values for LDAP Server:</p> <ul style="list-style-type: none"> <li><code>cn</code> (default for Microsoft AD)</li> <li><code>uid</code> (default for LDAP Server)</li> <li><code>distinguishedName</code></li> </ul> <div> <p><b>Note</b></p> <p>Whatever value you choose for this property, it should correspond to the one in the JSON transformation of your system (the <code>user</code> mapping).</p> </div>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>

**System Role:** Source

**System Role:** Source, Target, Proxy

Name	Description	System Type
<code>ldap.attribute.group.member</code>	Default value: <a href="#">member</a> <b>System Role:</b> Source, Target, Proxy	LDAP Server
<code>ldap.attribute.user.mail</code>	Default value: <a href="#">mail</a> <b>System Role:</b> Source, Target, Proxy	LDAP Server
<code>ldap.attribute.user.mobile</code>	Default value: <a href="#">mobile</a> <b>System Role:</b> Source, Target, Proxy	LDAP Server
<code>ldap.attribute.user.givenName</code>	Default value: <a href="#">givenName</a> <b>System Role:</b> Source, Target, Proxy	LDAP Server
<code>ldap.attribute.user.surname</code>	Default value: <a href="#">sn</a> <b>System Role:</b> Source, Target, Proxy	LDAP Server
<code>ldap.attribute.user.groups</code>	Default value: <a href="#">memberOf</a> <b>System Role:</b> Source, Target, Proxy	LDAP Server
<code>ldap.attribute.user.telephoneNumber</code>	Default value: <a href="#">telephoneNumber</a> <b>System Role:</b> Source, Target, Proxy	LDAP Server
<code>ldap.attribute.group.object.class.required</code>	<p>The LDAP object classes have attributes required for creation of entities.</p> <ul style="list-style-type: none"> <li>For Open LDAP Server, the required attribute is the common name (CN) of the group.</li> <li>For other implementations, it could be another attribute.</li> </ul> <p>Default value: <a href="#">cn</a></p> <p><b>System Role:</b> Target, Proxy</p>	LDAP Server
<code>ldap.attribute.user.object.class.required</code>	<p>The LDAP object classes have attributes required for creation of entities.</p> <ul style="list-style-type: none"> <li>For Open LDAP Server, the required attribute is the common name (CN) of the user.</li> <li>For other implementations, it could be another attribute.</li> </ul> <p>Default value: <a href="#">cn</a></p> <p><b>System Role:</b> Target, Proxy</p>	LDAP Server

Name	Description	System Type
<code>ldap.respond.with.resource.after.create</code>	<p>When set to <a href="#">true</a>, the SCIM create operation will read the created entity from the LDAP server.</p> <p>Value <a href="#">true</a> is required because the SCIM create operation must return the created entity.</p> <p>Default value: <a href="#">true</a></p> <p><b>System Role:</b> Proxy</p>	LDAP Server
<code>ldap.respond.with.resource.after.update</code>	<p>When set to <a href="#">true</a>, the SCIM update operation will read the modified entity from the LDAP server.</p> <p>When set to <a href="#">false</a> or the property is missing, the update operation will respond with error <a href="#">204</a> (no content).</p> <p>Default value: <a href="#">true</a></p> <p><b>System Role:</b> Proxy</p>	LDAP Server
<code>ldap.user.filter</code>	<p>You can optimize the search to return only particular users.</p> <p>To enter correct user filters, stick to the standard LDAP specification. See: <a href="#">LDAP Representation of Filters – Examples</a> .</p> <p><b>Possible values:</b></p> <p>For example:</p> <p>Value <a href="#">(cn=123*)</a> will return only users whose UID starts with "123" (such as <a href="#">1234567689</a> or <a href="#">1230011</a>).</p> <p>By default, this filter is empty. That is, if the property is not specified, the filter will search for every user.</p> <p><b>System Role:</b> Source</p>	<ul style="list-style-type: none"> <li>• LDAP Server</li> <li>• Microsoft Active Directory</li> </ul>

Name	Description	System Type
<code>ldap.group.filter</code>	<p>You can optimize the search to return only particular groups.</p> <p>To enter correct group filters, stick to the standard LDAP specification. See: <a href="#">LDAP Representation of Filters – Examples</a> .</p> <p><b>Possible values:</b></p> <p>For example:</p> <p>Value <code>(cn=mar*)</code> will return only groups whose CN starts with "mar" (such as <i>marked</i>, <i>March</i>, or <i>Marketing</i>).</p> <p>By default, this filter is empty. That is, if the property is not specified, the filter will search for every group.</p> <p><b>System Role:</b> Source</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<code>ldap.page.size</code>	<p>Use this property to configure the paging (pagination). That means, the number of entities to be read from the LDAP server at once.</p> <p><b>Possible values:</b></p> <p>Default value: <code>100</code></p> <div> <p><b>Note</b></p> <p>It is not recommended to exceed 1000.</p> </div> <p><b>System Role:</b> Source</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<code>ldap.api.version</code>	<p>Defines the version of LDAP Server API.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><b>1</b> - Indicates that LDAP Server API version 1 is used.</li> <li><b>2</b> - Indicates that LDAP Server API version 2 is used.</li> </ul> <p>If the property is not defined - LDAP Server API version 1 is used.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<p>LDAP Server</p>



Name	Description	System Type
<code>concur.page.size</code>	<p>Use this property to configure the paging. That means, the number of entities to be read from Concur at once.</p> <p><b>Possible values:</b></p> <p>Default value: <i>100</i></p> <div> <p><b>i Note</b></p> <p>The maximum allowed number is 100.</p> </div> <p><b>System Role:</b> Source</p>	SAP Concur
<code>msggraph-filter</code> <i>(Deprecated)</i>	<p>Use this property to filter users and groups by specific criteria, according to the API syntax of Microsoft Azure AD.</p> <div> <p><b>i Note</b></p> <p>This property is deprecated. Use <code>aad.user.filter</code> and <code>aad.group.filter</code> instead.</p> </div> <p><b>Possible values:</b></p> <p>Default value: <i>null</i></p> <p>To set a particular value, see <a href="#">Microsoft Graph: filter parameter</a> .</p> <p><b>System Role:</b> Source</p>	Microsoft Azure Active Directory
<code>gsuite.page.size</code>	<p>Use this property to configure the paging. That means, the number of entities to be read from Google G Suite at once.</p> <p><b>Possible values:</b></p> <p>Default value: <i>100</i></p> <div> <p><b>i Note</b></p> <p>The maximum allowed number is 500.</p> </div> <p><b>System Role:</b> Source</p>	Google G Suite

Name	Description	System Type
<code>gsuite.get.deleted</code>	<p>This property determines whether recently deleted entities should be read.</p> <div> <p><b>i Note</b></p> <p>You can apply this property only for <b>users</b>. For groups it will be ignored.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Source</p>	Google G Suite
<code>gsuite.domain</code>	<p>This property determines whether entities from a particular domain should be read.</p> <p><b>Possible values:</b></p> <p>For example:</p> <p><code>myaccount.ondemand.com</code></p> <p><b>System Role:</b> Source</p>	Google G Suite
<code>gsuite.customer.id</code>	<p>This property determines whether entities for a particular customer ID to be read. This property takes precedence over <code>gsuite.domain</code>.</p> <p><b>Possible values:</b> Customer ID number</p> <p>For more information, see <a href="#">Google G Suite API: User Accounts</a>.</p> <p><b>System Role:</b> Source</p>	Google G Suite
<code>com.sun.jndi.ldap.read.timeout</code>	<p>Use this property if you want to specify the read timeout (in milliseconds) for an LDAP connection.</p> <p><b>Possible values:</b></p> <p>For example: <code>5000</code></p> <p>This value causes the LDAP service provider to abort the read attempt if the server does not respond within 5 seconds.</p> <p><b>System Role:</b> Source</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>

Name	Description	System Type
<code>com.sun.jndi.ldap.connect.timeout</code>	<p>Use this property if you want to set the timeout (in milliseconds) for connecting to the LDAP server.</p> <p><b>Possible values:</b></p> <p>For example: <a href="#">500</a></p> <p>This value causes the LDAP service provider to abort the connection attempt if a connection cannot be established in half a second.</p> <p><b>System Role:</b> Source</p>	<ul style="list-style-type: none"> <li>LDAP Server</li> <li>Microsoft Active Directory</li> </ul>
<code>oauth.resource.name</code>	<p>Enter the URL to the Microsoft Graph.</p> <p><b>Possible values:</b> <a href="https://graph.microsoft.com">https://graph.microsoft.com</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	Microsoft Azure Active Directory

Name	Description	System Type
<code>ad.group.flatten</code>	<p>There are target systems that do not support nested groups (group structures). Therefore, if your Microsoft AD system contains such groups, they will not be resolved properly during the provisioning job. Such target systems are:</p> <ul style="list-style-type: none"> <li>• <a href="#">SAP Jam Collaboration</a></li> <li>• <a href="#">Identity Authentication</a></li> </ul> <p>To enable reading of group structures, you can use the <code>ad.group.flatten</code> property and set it to <a href="#">true</a>. It will read the group structure recursively and will "flatten" it so that all users from all groups and subgroups will be resolved and written in the target system as members of the main parent group.</p> <p>For best results, we recommend you also set the system property <code>ldap.group.filter</code> whose value is one or multiple Microsoft AD parent groups.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">true</a></li> <li>• <a href="#">false</a></li> </ul> <p>Default value: <a href="#">false</a></p> <p>Examples for filtering:</p> <ul style="list-style-type: none"> <li>• If your Microsoft AD system contains a parent group "<a href="#">Canteen</a>", which contains nested subgroups, you have to set the filter like this: <code>ldap.group.filter = (cn=<a href="#">Canteen</a>)</code> The Identity Provisioning will resolve all the direct users and groups of "<a href="#">Canteen</a>", along with all the users of its subgroups (and their subgroups). In the target system, all users will be written in one parent group named also "<a href="#">Canteen</a>".</li> <li>• If you have multiple parent groups (for example, <a href="#">Canteen</a>, <a href="#">Finances</a>,</li> </ul>	Microsoft Active Directory

Name	Description	System Type
	<p>and <a href="#">Support_Team</a>) that contain nested subgroups, you have to set the filter like this:</p> <pre>ldap.group.filter = (&amp;(cn=Canteen)(cn=Finance) (cn=Support_Team))</pre> <p><b>System Role:</b> Source</p>	
aad.domain.name	<p>Enter one of the verified domain names from the corresponding Azure AD tenant.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	Microsoft Azure Active Directory

Name	Description	System Type
<code>aad.group.member.attributes</code>	<p>This property defines the attributes of a group member to be read by the Identity Provisioning. By default, it always reads the <i>type</i> and the <i>id</i> of a member.</p> <p>If you prefer the Identity Provisioning to read additional attributes, you can add them as a single or a comma-separated value. For example:</p> <div> <p>❖ <b>Example</b></p> <ul style="list-style-type: none"> <li>If you want to read the e-mails too, enter:  <code>aad.group.member.attributes=mail</code>  This will read a member's type, ID and e-mail.</li> <li>If you want to read multiple additional attributes, enter:  <code>aad.group.member.attributes=mail,mobilePhone,displayName</code>  This will read a member's type, ID, e-mail, phone and display name.</li> </ul> </div> <p>See: <a href="#">Microsoft Azure Active Directory [page 685]</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>type</i> (default)</li> <li><i>id</i> (default)</li> <li>Any valid Microsoft Azure attribute of a group member</li> <li>A comma-separated list of valid MS Azure attributes of a group member</li> </ul> <div> <p>→ <b>Remember</b></p> <p>The Identity Provisioning service always retrieves the <i>id</i> and <i>type</i> attributes of a group member, regardless of the additional attributes you specify.</p> </div>	Microsoft Azure Active Directory
<b>System Role:</b> Source, Proxy		

Name	Description	System Type
<code>aad.user.attributes.membership.active</code>	<p>Use this property if you want to get information about all the groups to which the users are assigned (if any).</p> <ul style="list-style-type: none"> <li>If the property is missing, or is set to <code>false</code> – group membership details for the users will not be extracted.</li> <li>If the property is set to <code>true</code> – group membership details for the users will be extracted.</li> </ul> <p>If you set the property to <code>true</code>, you will get information about the group ID and its entity type (group) – default result. However, if you also set a value for property <code>aad.group.attributes</code>, you will get additional information relevant to this value.</p> <p>For example:</p> <p>If you set <code>aad.user.attributes.membership.active = true</code> and <code>aad.group.attributes = displayName</code>, you will receive the following exemplary data for a group as part of the user object:</p> <pre> "groups": [   {     "displayName": "Azure AD Group 1",     "id": "aa111999-0000-444-123-777fff000",     "type": "group"   } ] </pre> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code> (default)</li> </ul> <p><b>System Role:</b> Source, Proxy</p>	Microsoft Azure Active Directory

Name	Description	System Type
<code>aad.user.filter</code>	<p>Via this property, you can filter users by specific criteria, according to the syntax of <a href="#">Microsoft Graph REST API</a> .</p> <p>You can also filter out users with advanced query parameters, as described in <a href="#">Advanced query capabilities on Azure AD objects</a> .</p> <div> <p><b>Note</b></p> <p>This property replaces the deprecated <code>msgraph-filter</code> property.</p> </div> <p><b>Possible values:</b> Text/numeric string</p> <p>For example:</p> <ul style="list-style-type: none"> <li>Value = <code>Department eq 'Finance'</code></li> <li>Value = <code>displayName eq 'John Smith' and city eq 'Sofia'</code></li> <li>Value = <code>userPrincipalName ne 'Julie Armstrong'</code></li> </ul> <p><b>System Role:</b> Source, Proxy</p>	Microsoft Azure Active Directory



Name	Description	System Type
aad.group.filter	<p>Via this property, you can filter groups by specific criteria, according to the syntax of <a href="#">Microsoft Graph REST API</a> .</p> <p>You can also filter out groups with advanced query parameters, as described in <a href="#">Advanced query capabilities on Azure AD objects</a> .</p> <p><b>Possible values:</b> Text/numeric string</p> <p>For example:</p> <ul style="list-style-type: none"> <li>Value = <i>displayName eq 'Employees 2020'</i></li> <li>Value = <i>displayName eq 'Service Administrators' and mail eq 'serviceadmins@abcd.onmicrosoft.com'</i></li> <li>Value = <i>startsWith(displayName, 'ABC_')</i></li> <li>Value = <i>displayName ne 'Service Administrators'</i></li> </ul> <p><b>System Role:</b> Source, Proxy</p>	Microsoft Azure Active Directory

Name	Description	System Type
aad.user.attributes	<p>Defines which user attributes are read from Microsoft Azure AD system.</p> <p>The property is set during system creation with the following default value:  <a href="#">id,mail,userPrincipalName,displayName,mailNickname,givenName,surname,mobilePhone,businessPhones</a></p> <p>This means that by default, Identity Provisioning will read from MS Azure AD the user attributes defined in the property value. Those attributes are also used in the default read transformation.</p> <p>To check the complete set of user attributes (properties) supported by Microsoft Azure AD, see: <a href="#">Microsoft Graph: User Properties</a> ➔</p> <p>If you want the Identity Provisioning to read additional user attributes, add them to the default list of attributes separated by comma and adapt the transformations.</p> <p>For example, to read the <a href="#">employeeId</a> of the MS Azure AD users in addition to the default list of attributes, and provision them to Identity Authentication, proceed as follows:</p> <ol style="list-style-type: none"> <li>1. Add the attribute in the property value:  id,mail,userPrincipalName,displayName,mailNickname,givenName,surname,mobilePhone,businessPhones,<a href="#">employeeId</a></li> <li>2. Extend the MS Azure AD read transformation by adding the following mapping for the user resource:</li> </ol>	Microsoft Azure Active Directory


#### Sample Code

```
{
  "sourcePath":
    "$.employeeId",
```

Name	Description	System Type
	<pre>"targetPath": "\$ ['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['employeeNumber'] ", "optional": true },</pre>	
	<p>3. Make sure the following mapping is present in the Identity Authentication write transformation:</p> <pre>Sample Code</pre> <pre>{   "sourcePath": "\$ ['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['employeeNumber'] ",   "targetPath": "\$ ['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['employeeNumber'] ",   "optional": true },</pre> <p>In case you remove the default list of attributes from the value of this property and only add the additional attributes, Identity Provisioning will return the additional user attributes plus the mandatory ones: <i>id,mail, userPrincipalName</i>.</p> <p><b>System Role:</b> Source, Proxy</p>	

Name	Description	System Type
aad.group.attributes	<p>Defines which group attributes are read from Microsoft Azure AD system.</p> <p>The property is set during system creation with the following default value: <i>id,displayName,mailNickname</i></p> <p>This means that by default, Identity Provisioning will read from MS Azure AD the group attributes defined in the property value and will also return the <i>members</i> attribute. Those attributes are used in the default read transformation.</p> <p>To check the complete set of group attributes (properties) supported by Microsoft Azure AD, see: <a href="#">Microsoft Graph: Group Properties</a> ➔</p> <p>If you want the Identity Provisioning to read additional group attributes, add them to the default list of attributes separated by comma and adapt the transformations.</p> <p>For example, to read the <i>description</i> of the MS Azure AD groups in addition to the default list of attributes, and provision them to Identity Authentication, proceed as follows:</p> <ol style="list-style-type: none"> <li>1. Add the attribute in the property value: id,displayName,mailNickname,<i>description</i></li> <li>2. Extend the MS Azure AD read transformation by adding the following mapping for the group resource:</li> </ol> <div data-bbox="644 1621 992 1919"> <p>≡ Code Syntax</p> <pre>{   "sourcePath":     "\$.description",   "optional": true,   "targetPath":     "\$.description" },</pre> </div>	Microsoft Azure Active Directory

Name	Description	System Type
	<p>3. Extend the Identity Authentication write transformation by adding the following mapping for the group resource:</p> <div data-bbox="644 517 992 913" data-label="Code-Block"> <pre> ≡, Code Syntax {   "sourcePath":     "\$.description",     "optional": true,     "targetPath": "\$     [ 'urn:sap:cloud:sc     im:schemas:extensi     on:custom:2.0:Grou     p' ]     [ 'description' ]"   }, </pre> </div> <p>In case you remove the default list of attributes from the value of this property and only add the additional attributes, Identity Provisioning will read from MS Azure AD the additional group attributes, the group <i>id</i>, <i>displayName</i>, <i>mailNickname</i> and will also return the <i>members</i> attribute.</p> <p><b>System Role:</b> Source, Proxy</p>	
<code>aad.entities.top</code>	<p>This property defines the number of entities to be read per page.</p> <p>Default value: <i>100</i></p> <p><b>System Role:</b> Source, Proxy</p>	Microsoft Azure Active Directory
<code>csrf.token.path</code>	<p>Path added to the URL to retrieve the CSRF token.</p> <p>The property is automatically added in the system, with default value: <i>/api/v1/scim/Users?count=1</i>.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Analytics Cloud

Name	Description	System Type
s4hana.cloud.hr.switch.active	<p>A default property, whose only possible value is <i>true</i>. That means, HR integration is enabled for your system.</p> <div>  <b>Caution</b>  Do not change this value! Otherwise, your provisioning job will fail. </div> <p><b>Possible value:</b> <i>true</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP S/4HANA Cloud
s4hana.cloud.roles.filter	<p>Enter OData filtering for reading roles in the SAP S/4HANA Cloud system.</p> <p>To learn what criteria you can use, see:  <a href="#">OData URI Conventions</a> ➔ 4.5 Filter System Query Option</p> <p><b>System Role:</b> Source, Proxy</p>	SAP S/4HANA Cloud
s4hana.cloud.api.version	<p>This property defines the API version that your SAP S/4HANA Cloud system uses.</p> <p>Version <i>1</i> means your SAP S/4HANA Cloud system uses <i>SAP_COM_0193</i> communication arrangement.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP S/4HANA Cloud
s4hana.cloud.hr.switch.dependent.role.codes	<p>A default property.</p> <p>Add the codes of the roles maintained by the HR integration. Make sure these role codes are part of your read and write transformations.</p> <p><b>Possible values:</b></p> <p>For example: <i>BUP003</i>, <i>BBP005</i></p> <p>That means, your HR integration will support <i>employees</i> and <i>contingent worker</i>.</p> <p><b>System Role:</b> Target, Proxy</p>	SAP S/4HANA Cloud


Name	Description	System Type
<code>s4hana.cloud.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source SAP S/4HANA Cloud system, you can choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>In the source and proxy systems, this property is activated by default. If you want to always read disabled entities, set the property to <i>false</i>, or delete it.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>true</i></p> <p><b>System Role:</b> Source, Proxy</p>	SAP S/4HANA Cloud
<code>s4hana.onprem.hr.switch.dependent.role.codes</code>	<p>Add the codes of the roles maintained by the HR integration. Make sure these role codes are part of your read and write transformations.</p> <p>This property is applicable only if <code>s4hana.onprem.hr.switch.active</code> = <i>true</i></p> <p><b>Possible values:</b></p> <p>For example: <i>BUP003</i>, <i>BBP005</i>, <i>BUP012</i>, <i>WFM001</i></p> <p>That means, your HR integration will support <i>employees</i>, <i>contingent workers</i>, <i>collaboration users</i>, and <i>resources</i>.</p> <p><b>System Role:</b> Target, Proxy</p>	SAP S/4HANA On-Premise

Name	Description	System Type
<code>s4hana.onprem.hr.switch.active</code>	<p>Defines whether the system should include HR integration or not.</p> <p>This property is related to <code>s4hana.onprem.hr.switch.dependent.role.codes</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> – HR integration is enabled for your system</li> <li><code>false</code> – HR integration is disabled for your system</li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP S/4HANA On-Premise
<code>s4hana.onprem.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source SAP S/4HANA On-Premise system, you can choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>In the source and proxy systems, this property is activated by default. If you want to always read disabled entities, set the property to <code>false</code>, or delete it.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>true</code></p> <p><b>System Role:</b> Source, Proxy</p>	SAP S/4HANA On-Premise




Name	Description	System Type
<code>s4hana.onprem.sap-client</code>	<p>Use this property if you want to specify a particular AS ABAP client to use as the <code>sap-client</code> URL parameter.</p> <p>If this property is not specified, the URL will open your default AS ABAP client. To learn more, see: <a href="#">Specifying the Client</a></p> <p>For more information about <code>sap-client</code>, see: <a href="#">SAP URL Parameters</a></p> <p><b>Possible values:</b> A three-digit integer number</p> <p>For example: <a href="#">102</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP S/4HANA On-Premise
<code>s4hana.onprem.support.bulk.operation</code>	<p>Set this property to <a href="#">true</a> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request.</p> <p>For more information, see: <a href="#">APIs for Business User Management</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">true</a></li> <li><a href="#">false</a></li> </ul> <p>Default value: <a href="#">false</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP S/4HANA On-Premise
<code>s4hana.onprem.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">20</a></p> <p>Maximum value: <a href="#">100</a></p> <p>If you enter a number larger than 100, the service will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	SAP S/4HANA On-Premise

Name	Description	System Type
fg.group.prefix	<p>This property distinguishes SAP Fieldglass groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>FG_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Fieldglass source system and will be provisioned to the target system with the following name pattern: <b>FG_&lt;GroupDisplayName&gt;</b>. This way SAP Fieldglass groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Fieldglass groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>FG_</b> prefix in their display name will be provisioned to SAP Fieldglass. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Fieldglass.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP Fieldglass


Name	Description	System Type
<code>fg.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable bulk operations for provisioning users and groups. This means, the Identity Provisioning service can write, update, and delete multiple users in a single request.</p> <p>For more information, see: <a href="#">SAP Fieldglass Identity Management API</a> </p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP Fieldglass
<code>fg.bulk.operations.max.count</code>	<p>This property sets the number of operations to be performed in one bulk request.</p> <p><b>Possible values:</b></p> <p>Default value: <i>20</i></p> <p>Minimum value: <i>10</i></p> <p>Maximum value: <i>100</i></p> <p>If you provide a value outside of the minimum and maximum range, the service will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	SAP Fieldglass

Name	Description	System Type
<code>a4c.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source SAP BTP ABAP environment system, you can choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>In the source and proxy systems, this property is activated by default. If you want to always read disabled entities, set the property to <i>false</i>, or delete it.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>true</i></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP BTP ABAP environment
<code>a4c.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP BTP ABAP environment system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➡ 4.5 Filter System Query Option</p> <p><b>System Role:</b> Source, Proxy</p>	SAP BTP ABAP environment


Name	Description	System Type
<code>a4c.roles.prefix</code>	<p>This property distinguishes SAP BTP ABAP environment roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>A4C_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the SAP BTP ABAP environment source system and will be provisioned to the target system with the following name pattern: <b>A4C_&lt;role_name&gt;</b> . This way SAP BTP ABAP environment roles in the target system will be easily distinguished from roles provisioned from other applications. If the property is not set, the SAP BTP ABAP environment roles will be read and provisioned to the target system with their actual role names.</li> <li>When <b>set in the target system</b>, only roles containing the <b>A4C_</b> prefix in their role name will be provisioned to SAP BTP ABAP environment. Roles without this prefix in their names won't be provisioned. If the property is not set, all roles will be provisioned to SAP BTP ABAP environment.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP BTP ABAP environment

Name	Description	System Type
<code>a4c.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP BTP ABAP environment target or proxy system.</p> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li><code>false</code> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>Default value (if the property is missing during system creation): <code>true</code></p> <p>Default value (if the property appears during system creation): <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP BTP ABAP environment

Name	Description	System Type
<code>ibp.roles.prefix</code>	<p>This property distinguishes SAP Integrated Business Planning for Supply Chain roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>IBP_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the SAP Integrated Business Planning for Supply Chain source system and will be provisioned to the target system with the following name pattern: <b>IBP_&lt;role_name&gt;</b> . This way SAP Integrated Business Planning for Supply Chain roles in the target system will be distinguished from roles provisioned from other applications.</li> </ul> <p>If the property is not set, the SAP Integrated Business Planning for Supply Chain roles will be read and provisioned to the target system with their actual role names.</p> <ul style="list-style-type: none"> <li>When <b>set in the target system</b>, only roles containing the <b>IBP_</b> prefix in their role name will be provisioned to SAP Integrated Business Planning for Supply Chain. Roles without this prefix in the role name won't be provisioned.</li> </ul> <p>If the property is not set, all roles will be be provisioned to SAP Integrated Business Planning for Supply Chain.</p> <p><b>System Role:</b> Source and Target</p>	SAP Integrated Business Planning for Supply Chain

Name	Description	System Type
<code>ibp.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP IBP target or proxy system.</p> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li><code>false</code> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>Default value (if the property is missing during system creation): <code>true</code></p> <p>Default value (if the property appears during system creation): <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Integrated Business Planning for Supply Chain



Name	Description	System Type
<code>marketing.cloud.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP Marketing Cloud target or proxy system.</p> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li><code>false</code> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>Default value (if the property is missing during system creation): <code>true</code></p> <p>Default value (if the property appears during system creation): <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Marketing Cloud
<code>s4hana.cloud.roles.page.size</code>	<p>This property indicates how many business roles (considered as <a href="#">groups</a>) per page to be read from your SAP S/4HANA Cloud source system.</p> <p><b>Possible values:</b> Integer number</p> <p>For example, if you set the property's value = <code>30</code>, the Identity Provisioning will read 30 roles (groups) at once, then – another 30, and so on.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP S/4HANA Cloud


Name	Description	System Type
<code>s4hana.cloud.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request.</p> <p>For more information, see: <a href="#">APIs for Business User Management</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP S/4HANA Cloud
<code>s4hana.cloud.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p><b>Possible values:</b></p> <p>Default value: <i>20</i></p> <p>Maximum value: <i>100</i></p> <p>If you enter a number larger than 100, the service will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	SAP S/4HANA Cloud
<code>marketing.cloud.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request.</p> <p>For more information, see: <a href="#">APIs for Business User Management</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP Marketing Cloud

Name	Description	System Type
<code>marketing.cloud.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">20</a></p> <p>Maximum value: <a href="#">100</a></p> <p>If you enter a number larger than 100, the service will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	SAP Marketing Cloud
<code>a4c.support.bulk.operation</code>	<p>Set this property to <a href="#">true</a> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request.</p> <p>For more information, see: <a href="#">APIs for Business User Management</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">true</a></li> <li>• <a href="#">false</a></li> </ul> <p>Default value: <a href="#">false</a></p> <p><b>System Role:</b> Target</p>	SAP BTP ABAP environment
<code>a4c.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">20</a></p> <p>Maximum value: <a href="#">100</a></p> <p>If you enter a number larger than 100, the service will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	SAP BTP ABAP environment

Name	Description	System Type
<code>a4c.roles.page.size</code>	<p>This property indicates how many business roles (considered as <i>groups</i>) per page to be read from your SAP BTP ABAP environment source system.</p> <p><b>Possible values:</b> Integer number</p> <p>For example, if you set the property's value = <i>30</i>, the Identity Provisioning will read 30 roles (groups) at once, then – another 30, and so on.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP BTP ABAP environment
<code>ibp.roles.page.size</code>	<p>This property indicates how many business roles (considered as <i>groups</i>) per page to be read from your SAP IBP source system.</p> <p><b>Possible values:</b> Integer number</p> <p>For example, if you set the property's value = <i>30</i>, the Identity Provisioning will read 30 roles (groups) at once, then – another 30, and so on.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Integrated Business Planning for Supply Chain
<code>ibp.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request.</p> <p>For more information, see: <a href="#">APIs for Business User Management</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP Integrated Business Planning for Supply Chain

Name	Description	System Type
<code>ibp.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p><b>Possible values:</b></p> <p>Default value: <code>20</code></p> <p>Maximum value: <code>100</code></p> <p>If you enter a number larger than 100, the service will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	SAP Integrated Business Planning for Supply Chain
<code>marketing.cloud.roles.page.size</code>	<p>This property indicates how many business roles (considered as <i>groups</i>) per page to be read from your SAP Marketing Cloud source system.</p> <p><b>Possible values:</b> Integer number</p> <p>For example, if you set the property's value = <code>30</code>, the Identity Provisioning will read 30 roles (groups) at once, then – another 30, and so on.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Marketing Cloud

Name	Description	System Type
marketing.cloud.roles.prefix	<p>This property distinguishes SAP Marketing Cloud roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SMKC_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the SAP Marketing Cloud source system and will be provisioned to the target system with the following name pattern: <b>SMKC_&lt;role_name&gt;</b> . This way SAP Marketing Cloud roles in the target system will be distinguished from roles provisioned from other applications.</li> </ul> <p>If the property is not set, the SAP Marketing Cloud roles will be read and provisioned to the target system with their actual role names.</p> <ul style="list-style-type: none"> <li>When <b>set in the target system</b>, only roles containing the <b>SMKC_</b> prefix in their role name will be provisioned to SAP Marketing Cloud. Roles without this prefix in the role name won't be provisioned.</li> </ul> <p>If the property is not set, all roles will be provisioned to SAP Marketing Cloud.</p> <p><b>System Role:</b> Source and Target</p>	SAP Marketing Cloud

Name	Description	System Type
<code>s4hana.cloud.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP S/4HANA Cloud target or proxy system.</p> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li><code>false</code> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>Default value (if the property is missing during system creation): <code>true</code></p> <p>Default value (if the property appears during system creation): <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP S/4HANA Cloud
<code>ibp.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source SAP IBP system, you can choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>In the source systems, this property is activated by default. If you want to always read disabled entities, set the property to <code>false</code>, or delete it.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>true</code></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Integrated Business Planning for Supply Chain

Name	Description	System Type
<code>ibp.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP IBP system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ 4.5 Filter System Query Option</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Integrated Business Planning for Supply Chain
<code>marketing.cloud.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source SAP Marketing Cloud system, you can choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>In the source and proxy systems, this property is activated by default. If you want to always read disabled entities, set the property to <i>false</i>, or delete it.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>true</i></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Marketing Cloud
<code>marketing.cloud.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP Marketing Cloud system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ 4.5 Filter System Query Option</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Marketing Cloud
<code>scim.api.csrf.protection</code>	<p>Specifies whether to fetch a CSRF token when sending requests to the system. The property is automatically added in the system, with default value: <i>enabled</i>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>enabled</i></li> <li><i>disabled</i></li> </ul> <p>Default value: <i>enabled</i></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Analytics Cloud



Name	Description	System Type
<code>sac.api.csrf.protection</code>	<p>Specifies whether to fetch a CSRF token when sending requests to the system. The property is automatically added in the system, with default value: <i>enabled</i>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>enabled</i></li> <li><i>disabled</i></li> </ul> <p>Default value: <i>enabled</i></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Analytics Cloud
<code>sac.support.bulk.operation</code>	<p>This property enables bulk operations for users.</p> <p>When bulk operations are enabled (set to <i>true</i>), Identity Provisioning service creates, updates, and deletes multiple users in one request.</p> <p>When bulk operations are not enabled (set to <i>false</i>), Identity Provisioning service creates, updates, and deletes one user at a time.</p> <p>For more information, see: <a href="#">SCIM Protocol: Bulk Operations</a> ➔</p> <div> <p><b>Note</b></p> <p>SCIM bulk operations are not supported for provisioning groups to SAP Analytics Cloud.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP Analytics Cloud

Name	Description	System Type
sac.bulk.operations.max.count	<p>If you have enabled the SCIM bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p><b>Possible values:</b></p> <p>Default value:</p> <ul style="list-style-type: none"> <li>100 - when using SAP Analytics Cloud SCIM API version 1.</li> <li>30 - when using SAP Analytics Cloud SCIM API version 2.</li> </ul> <div> <p><b>Note</b></p> <p>The value must not exceed the number of entities defined by the SAP Analytics Cloud system as a SCIM service provider. Otherwise, the provisioning job will fail with HTTP response code 413 (<i>Payload Too Large</i>).</p> </div> <p><b>System Role:</b> Target</p>	SAP Analytics Cloud
scim.user.filter	<p>When specified, only those users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example:</p> <p><i>name.familyName eq "SmithJ" and addresses.country eq "US"</i></p> <p><b>System Role:</b> Source</p>	<ul style="list-style-type: none"> <li>SCIM System</li> <li>SAP Analytics Cloud</li> <li>SAP Commissions</li> <li>SAP Jam Collaboration</li> <li>Identity Authentication (SCIM API version 1)</li> <li>Local Identity Directory</li> <li>Cloud Foundry UAA Server</li> <li>SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>Sales Cloud – Analytics &amp; AI</li> <li>SAP BTP Account Members (Neo)</li> <li>SAP Fieldglass</li> </ul>

Name	Description	System Type
<code>scim.group.filter</code>	<p>When specified, only those groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example:</p> <p><i>displayName eq "ProjectTeam1" or "Students2018"</i></p> <p><b>System Role:</b> Source</p>	<ul style="list-style-type: none"> <li>• SCIM System</li> <li>• SAP Analytics Cloud</li> <li>• SAP Commissions</li> <li>• SAP Jam Collaboration</li> <li>• Local Identity Directory</li> <li>• Cloud Foundry UAA Server</li> <li>• SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>• Sales Cloud – Analytics &amp; AI</li> <li>• SAP BTP Account Members (Neo)</li> <li>• SAP Fieldglass</li> </ul>
<code>scim.content.type</code>	<p>Makes the connector send a specified value for the <i>Content-Type</i> HTTP header. This is needed because a SCIM system could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>If the property is not specified, the default value is taken: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	<ul style="list-style-type: none"> <li>• SCIM System</li> <li>• SAP Analytics Cloud</li> <li>• SAP Commissions</li> <li>• SAP Jam Collaboration</li> <li>• Identity Authentication (SCIM API version 1)</li> <li>• Local Identity Directory</li> <li>• Cloud Foundry UAA Server</li> <li>• SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>• Sales Cloud – Analytics &amp; AI</li> <li>• SAP BTP Account Members (Neo)</li> <li>• SAP Fieldglass</li> </ul>

Name	Description	System Type
<code>scim.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). <b>If the service finds such a user on the target system via this filter, then the conflicting user will overwrite the existing one.</b> If the service does not find such a user, the creation will fail.</p> <p>According to your use case and system type, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> </ul>	<ul style="list-style-type: none"> <li>SCIM System</li> <li><a href="#">SAP Analytics Cloud</a></li> <li>SAP Commissions</li> <li>SAP Jam Collaboration</li> <li><a href="#">Identity Authentication</a> (SCIM API version 1)</li> <li>Local Identity Directory</li> <li>Local Identity Directory (when Identity Provisioning is running on SAP Cloud Identity Infrastructure)</li> <li>Cloud Foundry UAA Server</li> <li>SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>Sales Cloud – Analytics &amp; AI</li> <li>SAP BTP Account Members (Neo)</li> <li>SAP Fieldglass</li> </ul>

**i Note**

Relevant only for [Identity Authentication](#) and [SAP Analytics Cloud](#):

- For systems created **before April 7, 2020**, this property is missing during system creation, and it has the default value, *userName*. If the service does not find an existing user with such a *userName*, it will try again to resolve the conflicting user – by *email*. If the second attempt for resolution is unsuccessful too, the creation of the conflicting user fails.

Name	Description	System Type
	<ul style="list-style-type: none"> <li>For systems created <b>after April 7, 2020</b>, this property appears by default during system creation, and its value is set to <i>userName</i>. If the service does not find an existing user with such a <i>userName</i>, the creation of the conflicting user fails. However, if you delete the property, the service will try again to resolve the conflicting user – by <i>email</i>. If the second attempt for resolution is unsuccessful too, the creation of the conflicting user fails.</li> <li>Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>userName</i></li> <li><i>emails[0].value</i></li> <li><i>userName,emails[0].value</i></li> <li><i>externalId</i>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p><b>Note</b></p> <ul style="list-style-type: none"> <li><i>phoneNumbers[0].value</i> - supported unique attribute for Lo-</li> </ul>	

Name	Description	System Type
	<p>cal Identity Directory (when Identity Provisioning is running on SAP Cloud Identity Infrastructure)</p> <p>Default value: <i>userName</i></p> <p><b>System Role:</b> Target</p>	
<code>scim.group.unique.attribute</code>	<p>If the service tries to create a group that already exists in the target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific).</p> <p>To make the search filter by a specific attribute, specify this attribute as a value for the <code>scim.group.unique.attribute</code> property.</p> <p>If the property is not specified, the search is done by the default attribute: <i>displayName</i></p> <p><b>System Role:</b> Target, Proxy</p>	<ul style="list-style-type: none"> <li>• SCIM System</li> <li>• SAP Analytics Cloud</li> <li>• SAP Commissions</li> <li>• SAP Jam Collaboration</li> <li>• Identity Authentication (SCIM API version 1)</li> <li>• Local Identity Directory</li> <li>• Cloud Foundry UAA Server</li> <li>• SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>• Sales Cloud – Analytics &amp; AI</li> <li>• SAP BTP Account Members (Neo)</li> <li>• SAP Fieldglass</li> </ul>

Name	Description	System Type
<code>scim.group.members.additional.attributes</code>	<p>Defines additional attributes you can request from an <b>Identity Authentication</b> source system when reading groups.</p> <p>If you read groups through REST API, use the <code>GET</code> request. Add the additional attributes (coma-separated) as a value of the URL parameter <code>membersAdditionalAttributes</code>.</p> <p><b>Possible values:</b> a coma-separated list of attribute names</p> <p>You can add the following attributes:</p> <ul style="list-style-type: none"> <li>• <a href="#">emails</a></li> <li>• <a href="#">userName</a></li> <li>• <a href="#">displayName</a></li> <li>• <a href="#">urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber</a></li> </ul> <p><b>System Role:</b> Source</p>	Identity Authentication (SCIM API version 1)
<code>scim.include.if.match.wildcard.header</code>	<p>Makes the connector send the <code>If-Match</code> HTTP header with a value of "*" for every request to the target system. This header could be used by a SCIM system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">true</a></li> <li>• <a href="#">false</a></li> </ul> <p>Default value: <a href="#">false</a></p> <p><b>System Role:</b> Target, Proxy</p>	<ul style="list-style-type: none"> <li>• SCIM System</li> <li>• SAP Analytics Cloud</li> <li>• SAP Commissions</li> <li>• SAP Jam Collaboration</li> <li>• Identity Authentication (SCIM API version 1)</li> <li>• Local Identity Directory</li> <li>• Cloud Foundry UAA Server</li> <li>• SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>• Sales Cloud – Analytics &amp; AI</li> <li>• SAP BTP Account Members (Neo)</li> <li>• SAP Fieldglass</li> </ul>

Name	Description	System Type
<code>scim.support.patch.operation</code>	<p>If your target or proxy system is among the SCIM-based ones listed under <a href="#">System Type</a> and supports <code>PATCH</code> operations, set this property to <code>true</code>. This way, when the Identity Provisioning identifies a changed entity in the source system, it will execute the updates as <code>PATCH</code> requests instead of <code>PUT</code>. That means, only the changes will be written in the target system, instead of provisioning the whole entity data.</p> <p>Note that only attributes without <code>"scope"</code> in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with scope, such as:</p> <pre> {   "constant": "xuaa-dummy-value",   "targetPath": "\$id",   "scope": "createEntity" }</pre>	<ul style="list-style-type: none"> <li>• SAP Jam Collaboration</li> <li>• Local Identity Directory</li> <li>• Cloud Foundry UAA Server</li> <li>• SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>• SCIM System</li> </ul>
<p><b>Additional Information:</b></p> <p>There are different cases when an entity should be updated in the target system:</p> <ul style="list-style-type: none"> <li>• In the source system, some of the entity attributes have been changed, or new attributes have been added.</li> <li>• In the source system, a condition or a filter is set for this entity not to be read anymore.</li> <li>• The whole entity has been deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your</p>		



Name	Description	System Type
	<p>target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <b>deleteEntity</b>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	

Name	Description	System Type
jam.group.prefix	<p>This property distinguishes SAP Jam Collaboration groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SJC_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Jam Collaboration source system and will be provisioned to the target system with the following name pattern: <b>SJC_&lt;GroupDisplayName&gt;</b>. This way SAP Jam Collaboration groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Jam Collaboration groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>SJC_</b> prefix in their display name will be provisioned to SAP Jam Collaboration. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Jam Collaboration.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP Jam Collaboration

Name	Description	System Type
scp.user.userbase	<p>This property specifies the host to the identity provider to be used with this target system. All provisioned users can be authenticated only by this identity provider.</p> <p>If you use another IdP, enter its value as configured in the SAP BTP cockpit. For example:</p> <p>&lt;account_ID&gt;.accounts.ondemand.com</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">account.sap.com</a></p> <p><b>System Role:</b> Target, Proxy</p>	SAP BTP Account Members (Neo)
AuthType	<p>Enter the type of authentication used for access token retrieval for OAuth HTTP destinations.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Basic</a></li> <li>• <a href="#">Form</a></li> </ul> <p>Default value: <a href="#">Basic</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>• SCIM System</li> <li>• SAP Analytics Cloud</li> <li>• SAP Commissions</li> <li>• SAP Jam Collaboration</li> <li>• Identity Authentication</li> <li>• Local Identity Directory</li> <li>• Cloud Foundry UAA Server</li> <li>• SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>• Sales Cloud – Analytics &amp; AI</li> <li>• SAP BTP Account Members (Neo)</li> <li>• SAP Fieldglass</li> </ul>
CloudConnectorLocationId	<p>Relevant when the ProxyType property is set to <a href="#">OnPremise</a>. Use it only if your SAP Business Technology Platform account uses more than one Cloud Connector.</p> <p><b>Possible values:</b> String</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>• SSH Server (Beta)</li> <li>• SAP HANA Database (Beta)</li> <li>• LDAP Server</li> <li>• Microsoft Active Directory</li> <li>• All HTTP systems</li> </ul>
hana.jdbc.db.user	<p>Name of the SAP HANA Database user</p> <p><b>System Role:</b> Target</p>	SAP HANA Database (Beta)
hana.jdbc.db.password	<p>(Credential)</p> <p><b>System Role:</b> Target</p>	SAP HANA Database (Beta)

Name	Description	System Type
hana.jdbc.db.host	SAP HANA Database host  <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.db.port	SAP HANA Database port  <b>Possible values:</b> <a href="#">30015</a>  <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.access.type	<p>There are three types of SAP HANA access:</p> <ul style="list-style-type: none"> <li><a href="#">direct</a> – It requires only hana.jdbc.db.* properties</li> <li><a href="#">ssh.tunnel</a> – it requires hana.jdbc.db.* and hana.jdbc.ssh.tunnel.* properties.</li> <li><a href="#">cf.app.ssh.tunnel</a> – It requires hana.jdbc.ssh.tunnel.cf.* properties to establish an SSH tunnel to the Cloud Foundry application, from which to access the JDBC SQL port of SAP HANA.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">direct</a></li> <li><a href="#">ssh.tunnel</a></li> <li><a href="#">cf.app.ssh.tunnel</a></li> </ul> <p><b>System Role:</b> Target</p>	SAP HANA Database (Beta)
hana.jdbc.ssh.tunnel.username	The username used for opening the SSH Tunnel  <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.ssh.tunnel.cf.technical.user.origin	<p>This is the origin of the Cloud Foundry technical user, specified in property hana.jdbc.ssh.tunnel.cf.username.</p> <p>If the origin is the same as of the other Cloud Foundry users, you don't need this property – leave it empty or delete it.</p> <p><b>Possible values:</b> Text/numeric string</p> <p>For example: <a href="#">uaa</a></p> <p><b>System Role:</b> Target</p>	SAP HANA Database (Beta)

Name	Description	System Type
hana.jdbc.ssh.tunnel.host	SSH Tunnel's host <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.ssh.tunnel.port	SSH Tunnel's port <b>Possible values:</b> <a href="#">22</a> <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.ssh.tunnel.auth.type	The authentication type for the SSH Tunnel. <b>Possible values:</b> Supported SSH authentication types: <ul style="list-style-type: none"> <li>• <a href="#">key</a></li> <li>• <a href="#">pwd</a></li> <li>• <a href="#">otp</a></li> <li>• <a href="#">key+otp</a></li> <li>• <a href="#">key+pwd</a></li> <li>• <a href="#">pwd+otp</a></li> <li>• <a href="#">key+pwd+otp</a></li> </ul> <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.ssh.tunnel.cf.api.url	The URL of the Cloud Foundry API. <b>Possible values:</b> For example: <a href="#">https://api.cf.mycloudfoundryhost.ondemand.com</a> <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.ssh.tunnel.cf.oauth.token.url	The URL of the OAuth token endpoint. <div> <b>→ Remember</b>  Remove the <a href="#">/oauth/token</a> part at the end of the URL. </div> <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.ssh.tunnel.cf.org	This is the Cloud Foundry organization. <b>System Role:</b> Target	SAP HANA Database (Beta)
hana.jdbc.ssh.tunnel.cf.space	This is the Cloud Foundry space. <b>System Role:</b> Target	SAP HANA Database (Beta)

Name	Description	System Type
<code>hana.jdbc.ssh.tunnel.cf.app</code>	This is the Cloud Foundry application to which the <a href="#">SAP HANA Database (Beta)</a> system opens an SSH tunnel. For more information, see: <a href="#">Cloud Foundry: Accessing apps with SSH</a> 🖱️	SAP HANA Database (Beta)
	<b>System Role:</b> Target	
<code>hana.jdbc.ssh.tunnel.cf.app.instance</code>	This is the instance number of the Cloud Foundry application.	SAP HANA Database (Beta)
	<b>System Role:</b> Target	
<code>hana.jdbc.ssh.tunnel.cf.username</code>	This is the Cloud Foundry user. It has the role <b>Developer</b> for the space where the application is deployed.	SAP HANA Database (Beta)
	<b>System Role:</b> Target	
<code>hana.jdbc.ssh.tunnel.cf.password</code>	(Credential) The password for property <code>hana.jdbc.ssh.tunnel.cf.username</code>	SAP HANA Database (Beta)
	<b>System Role:</b> Target	
<code>hana.jdbc.ssh.tunnel.password</code>	(Credential) Taken into account only if the authentication type includes <b>pwd</b> . That means any of the following: <ul style="list-style-type: none"> <li><code>hana.jdbc.ssh.tunnel.auth.type = <a href="#">pwd</a></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <a href="#">pwd+otp</a></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <a href="#">key+pwd</a></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <a href="#">key+pwd+otp</a></code></li> </ul>	SAP HANA Database (Beta)
	<b>System Role:</b> Target	

Name	Description	System Type
<code>hana.jdbc.ssh.tunnel.totp.secret.key</code>	<p>(Credential) Taken into account only if the authentication type includes <b>otp</b>. That means any of the following:</p> <ul style="list-style-type: none"> <li><code>hana.jdbc.ssh.tunnel.auth.type = otp</code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = key+otp</code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = pwd+otp</code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = key+pwd+otp</code></li> </ul> <p><b>System Role:</b> Target</p>	SAP HANA Database (Beta)
<code>hana.jdbc.ssh.tunnel.private.key</code>	<p>(Credential) Taken into account only if the authentication type includes <b>key</b>. That means any of the following:</p> <ul style="list-style-type: none"> <li><code>hana.jdbc.ssh.tunnel.auth.type = key</code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = key+pwd</code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = key+otp</code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = key+pwd+otp</code></li> </ul> <p><b>System Role:</b> Target</p>	SAP HANA Database (Beta)
<code>sales.cloud.analytics_ai.user.filter</code>	<p>Use this property to filter users by specific criteria, according to the API syntax of SCAAI.</p> <p><b>Possible values:</b> Text/numeric string</p> <p>For example: <code>externalId eq "John123"</code></p> <p><b>System Role:</b> Source, Proxy</p>	Sales Cloud – Analytics & AI
<code>sales.cloud.analytics_ai.group.filter</code>	<p>Use this property to filter groups by specific criteria, according to the API syntax of SCAAI.</p> <p><b>Possible values:</b> Text/numeric string</p> <p>For example: <code>displayName eq "first_group"</code></p> <p><b>System Role:</b> Source, Proxy</p>	Sales Cloud – Analytics & AI

Name	Description	System Type
ssh.read.users.command	Path to the bash command you need to execute to read users. <b>System Role:</b> Source	SSH Server (Beta)
ssh.create.user.command	Path to the bash command you need to execute to create a user. <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.update.user.command	Path to the bash command you need to execute to update a user. <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.delete.user.command	Path to the bash command you need to execute to delete a user. <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.read.groups.command	Path to the bash command you need to execute to read groups. <b>System Role:</b> Source	SSH Server (Beta)
ssh.create.group.command	Path to the bash command you need to execute to create a group. <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.update.group.command	Path to the bash command you need to execute to update a group. <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.delete.group.command	Path to the bash command you need to execute to delete a group. <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.create.user.command.exit.code.already.exists	An exit code number <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.update.user.command.exit.code.not.found	An exit code number <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.delete.user.command.exit.code.not.found	An exit code number <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.create.group.command.exit.code.already.exists	An exit code number <b>System Role:</b> Source, Target	SSH Server (Beta)



Name	Description	System Type
ssh.update.group.command.exit.code.not.found	An exit code number <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.delete.group.command.exit.code.not.found	An exit code number <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.auth.type	Supported SSH authentication types: <ul style="list-style-type: none"> <li>• <a href="#">key</a></li> <li>• <a href="#">pwd</a></li> <li>• <a href="#">otp</a></li> <li>• <a href="#">key+otp</a></li> <li>• <a href="#">key+pwd</a></li> <li>• <a href="#">pwd+otp</a></li> <li>• <a href="#">key+pwd+otp</a></li> </ul> <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.host	<b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.port	<b>Possible values:</b> <a href="#">22</a> <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.username	<b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.password	(Credential) Taken into account only if the authentication type includes <b>pwd</b> . That means any of the following: <ul style="list-style-type: none"> <li>• <code>ssh.auth.type = <a href="#">pwd</a></code></li> <li>• <code>ssh.auth.type = <a href="#">pwd+otp</a></code></li> <li>• <code>ssh.auth.type = <a href="#">key+pwd</a></code></li> <li>• <code>ssh.auth.type = <a href="#">key+pwd+otp</a></code></li> </ul> <b>System Role:</b> Source, Target	SSH Server (Beta)
ssh.totp.secret.key	(Credential) Taken into account only if the authentication type includes <b>otp</b> . That means any of the following: <ul style="list-style-type: none"> <li>• <code>ssh.auth.type = <a href="#">otp</a></code></li> <li>• <code>ssh.auth.type = <a href="#">key+otp</a></code></li> <li>• <code>ssh.auth.type = <a href="#">pwd+otp</a></code></li> <li>• <code>ssh.auth.type = <a href="#">key+pwd+otp</a></code></li> </ul> <b>System Role:</b> Source, Target	SSH Server (Beta)

Name	Description	System Type
<code>ssh.private.key</code>	<p>(Credential) Taken into account only if the authentication type includes <b>key</b>. That means any of the following:</p> <ul style="list-style-type: none"> <li><code>ssh.auth.type = key</code></li> <li><code>ssh.auth.type = key+pwd</code></li> <li><code>ssh.auth.type = key+otp</code></li> <li><code>ssh.auth.type = key+pwd+otp</code></li> </ul> <p><b>System Role:</b> Source, Target</p>	SSH Server (Beta)
<code>ssh.private.key.type</code>	<p>The format of SSH private key.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>ssh-rsa</code></li> <li><code>ssh-dsa</code></li> </ul> <p>Default value: <code>ssh-rsa</code></p> <p><b>System Role:</b> Source, Target</p>	SSH Server (Beta)
<code>sf.page.size</code>	<p>Use this property to configure the paging. That means, the number of entities to be read from SAP SuccessFactors at once.</p> <p>Default value: <code>100</code></p> <p><b>System Role:</b> Source, Proxy</p>	SAP SuccessFactors (using version 1 - SAP SuccessFactors HCM Suite OData API)

Name	Description	System Type
<code>sf.group.filter</code>	<p>The possible values of this property depend on the API version which your SAP SuccessFactors system consumes.</p> <p>Use this property to filter dynamic groups from SAP SuccessFactors. The filter obtains values as described in the OData 2.0 syntax, except any statements with attribute <code>lastModifiedDateTime</code>. To learn more, see:</p> <ul style="list-style-type: none"> <li>• <a href="#">OData version 2</a> ➔ <b>4.5. Filter System Query Option (\$filter)</b></li> <li>• <a href="#">SAP SuccessFactors HCM Suite OData API</a> ➔ <b>DynamicGroup</b></li> <li>• <a href="#">syntax, except anySAP SuccessFactors Workforce SCIM API</a></li> </ul> <p><b>Possible values:</b></p> <p>If your system consumes SAP SuccessFactors Workforce SCIM API, you can filter groups only by <code>displayName</code>.</p> <p>For example: <code>groupType eq 'permission'</code></p> <p><b>System Role:</b> Source, Proxy</p>	<ul style="list-style-type: none"> <li>• SAP SuccessFactors (using version 1 - SAP SuccessFactors HCM Suite OData API)</li> <li>• SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)</li> </ul>

Name	Description	System Type
<code>sf.group.prefix</code>	<p>This property distinguishes SAP SuccessFactors groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SF_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP SuccessFactors source system and will be provisioned to the target system with the following name pattern: <b>SF_&lt;GroupDisplayName&gt;</b>. This way SAP SuccessFactors groups in the target system will be distinguished from groups provisioned from other applications.</li> <li>When <b>set in the target system</b>, only groups containing the <b>SF_</b> prefix in their display name will be provisioned to SAP SuccessFactors. Groups without this prefix in the display name won't be provisioned.</li> </ul> <p>If the property is not set, the SAP SuccessFactors groups will be read and provisioned to the target system with their actual display names.</p> <p><b>System Role:</b> Source, Target</p>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)

Name	Description	System Type
<code>sf.user.filter</code>	<p>The possible values of this property depend on the API version which your SAP SuccessFactors system consumes.</p> <p>This property takes values as described in the <a href="#">OData version 2</a> <code>lastModifiedDateTime</code>.</p> <div> <p><b>⚠ Caution</b></p> <p>Attribute <code>lastModifiedDateTime</code> is used internally by the Identity Provisioning service, for calculating the delta load from the SAP SuccessFactors system. You must not use it in your filter statements. If you do, your provisioning jobs will fail.</p> </div> <div> <p><b>→ Tip</b></p> <p>By default, only <b>active</b> users are read from SAP SuccessFactors. If you want to filter by another user status, you can set it in the value of this property, using either the <a href="#">status value</a> or the <a href="#">status text</a> parameters. See:</p> <p><a href="#">SAP SuccessFactors HCM Suite OData API</a></p> <p><b>→ Querying Different Types of Users</b></p> </div> <p><b>Possible values:</b></p> <p>For example: <code>division eq 'Manufacturing (MANU)'</code></p> <div> <p><b>i Note</b></p> <p>You can only use attributes supported as filterable by the SAP SuccessFactors HCM Suite OData API.</p> <p>Some of these filterable attributes are: <code>firstName</code>, <code>lastName</code>, <code>department</code>, <code>division</code>,</p> </div>	<ul style="list-style-type: none"> <li>SAP SuccessFactors (using version 1 - SAP SuccessFactors HCM Suite OData API)</li> <li>SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)</li> </ul>

Name	Description	System Type
	<div> <code>jobCode, location, status, userId, username.</code> </div> <p>If your system consumes SAP SuccessFactors Workforce SCIM API, you can filter users by <code>userName</code>.</p> <p>For example: <i><code>userName eq "Sebastian"</code></i></p> <p>See : <a href="#">SAP SuccessFactors Workforce SCIM API</a></p> <p><b>System Role:</b> Source, Proxy</p>	

Name	Description	System Type
<code>sf.user.attributes</code>	<p>The value of this property is a comma-separated list of user attributes that have to be loaded from/to the SAP SuccessFactors system.</p> <p><b>Possible values:</b></p> <p>Default value:  <a href="#"><code>userId,username,status,email,lastName,firstName,lastModifiedDateTime,personKeyNav</code></a></p> <p>SAP SuccessFactors supports a huge amount of user information, which requires a lot of memory processing time and may even lead to time-out errors. That's why we recommend that you keep the default list of attributes, or specify only a few (the most significant attributes) for your provisioning scenario.</p> <div> <p><b>Note</b></p> <p>If you want to add more attributes, make sure you have added:</p> <ul style="list-style-type: none"> <li>the relevant extra attributes to the value of this property, separated by commas</li> <li>extra mappings for these attributes in the <a href="#">user</a> transformation</li> <li>extra mappings for these attributes in the write transformation of the relevant target system</li> </ul> </div> <div> <p><b>→ Remember</b></p> <p>Always make sure that attribute <code>lastModifiedDateTime</code> is in the list. If you don't specify it, the provisioning from/to SAP SuccessFactors will fail.</p> </div>	SAP SuccessFactors (using version 1 - SAP SuccessFactors HCM Suite OData API)
<b>System Role:</b> Source, Target, Proxy		

Name	Description	System Type
<code>sf.user.attributes.expand</code>	<p>This property reads/writes additional user data related to <a href="#">complex (navigation)</a> attributes, which are specified in the <code>sf.user.attributes</code> property.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">personKeyNav</a>, <a href="#">personKeyNav/</a> <a href="#">userAccountNav</a></p> <p>For example: If you also need to read the <a href="#">username</a> of the manager of a company employee, enter the following configuration in the <a href="#">Properties</a> tab:</p> <pre>sf.user.attributes = <a href="#">username</a>,<a href="#">firstName</a>,<a href="#">lastName</a>,<a href="#">manager</a> /<a href="#">username</a></pre> <pre>sf.user.attributes.expand = <a href="#">personKeyNav</a>,<a href="#">personKeyNav/</a> <a href="#">userAccountNav</a>,<a href="#">manager</a></pre> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP SuccessFactors (using version 1 - SAP SuccessFactors HCM Suite OData API)



Name	Description	System Type
<code>aad.user.attributes.expand</code>	<p>This property allows you to expand the list of attributes specified in the <code>aad.user.attributes</code> property with additional user attributes.</p> <p>Once you provide the value of the additional attributes in the <code>aad.user.attributes.expand</code> property, you need to extend the read transformation of MS Azure AD with attribute mappings based on the given value.</p> <p>Currently, the read transformation of MS Azure AD is extended with the attribute mappings for manager <b>id</b> and <b>displayName</b> as follows:</p> <pre> ≡ Code Syntax {   "sourcePath":     "\$.manager.id",    "targetPath": "\$[     'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'   ]['manager']     ['value']",     "optional":       true     },     {       "sourcePath":         "\$.manager.displayName",        "targetPath": "\$[         'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'       ]['manager']         ['displayName']",         "optional":           true         }       }     }   </pre> <p>To read the manager of the user, you need to provide the manager as a value of the <code>aad.user.attributes.expand</code></p>	Microsoft Azure Active Directory

Name	Description	System Type
	<p>property in the following format:</p> <p><b>manager(\$select=id,displayName)</b></p> <p>For more information on the attributes (relationships) that support the <b>\$expand</b> query parameter, refer to <a href="#">Microsoft Graph REST API v1.0</a> → <i>Relationships</i>.</p> <p><b>System Role:</b> Source, Proxy</p>	
aad.group.attributes.expand	<p>This property allows you to expand the list of attributes specified in the aad.group.attributes property with additional group attributes.</p> <p>Once you provide the value of the additional attributes in the aad.group.attributes.expand property, you need to extend the read transformation of MS Azure AD with attribute mappings based on the given value.</p> <p>For more information on the attributes (relationships) that support the <b>\$expand</b> query parameter, refer to <a href="#">Microsoft Graph REST API v1.0</a> → <i>Relationships</i>.</p> <p><b>System Role:</b> Source, Proxy</p>	Microsoft Azure Active Directory
RemoteSystemID	<div> <div><b>i Note</b></div> <div>Only relevant to API v.1.</div> </div> <p>Enter the system instance ID, configured for the communication system setting in the SAP Sales Cloud and SAP Service Cloud system.</p> <p><b>Possible values:</b></p> <p>For example: <i>IPS</i></p> <p><b>System Role:</b> Target</p>	SAP Sales Cloud and SAP Service Cloud

Name	Description	System Type
RecipientPartyID	<div> <i>Note</i>  Only relevant to API v.2. </div> <p>Enter the recipient system name.</p> <p><b>Possible values:</b></p> <p>For example: <i>0011SAP</i></p> <p><b>System Role:</b> Target</p>	SAP Sales Cloud and SAP Service Cloud
SenderPartyID	<div> <i>Note</i>  Only relevant to API v.2. </div> <p>Enter the name of the sender system name. It's equal to the value of property RemoteSystemID from API v.1.</p> <p><b>Possible values:</b></p> <p>For example: <i>IPS</i></p> <p><b>System Role:</b> Target</p>	SAP Sales Cloud and SAP Service Cloud

Name	Description	System Type
TrustAll	<p>Use this property when you create a connectivity destination in SAP BTP cockpit with authentication type <code>BasicAuthentication</code> to configure your provisioning system. Use cases:</p> <ul style="list-style-type: none"> <li>If this property is not specified or set to <code>false</code>, you need to add a truststore certificate to check for SSL connections. You can either use the default JDK truststore, or provide a custom truststore certificate – if you use a custom domain instead of the default Identity Authentication one. For more information, see: <a href="#">Use Destination Certificates (Cockpit)</a> <a href="#">Use Custom Domain in Identity Authentication</a></li> <li>If the property is enabled (set to <code>true</code>), the server certificate will be ignored, thus – not checked for SSL connections.</li> </ul> <div> <p>→ Remember</p> <p>For productive scenarios, we recommend that you do not use this property (or set it to <code>false</code>) because the SSL server certificate cannot be verified, and thus the server is not authenticated.</p> <p>Enable the property only for testing purposes.</p> </div>	All systems
<p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Source, Target, Proxy</p>		

Name	Description	System Type
<code>uaa.origin</code>	<p>It denotes the <code>origin</code> attribute in the system transformation.</p> <p>The value of this property is the location of your Cloud Foundry identity provider. If not sure about the value, ask your Cloud Foundry system administrator.</p> <p><b>Possible values:</b> Text/numeric string</p> <p><b>System Role:</b> Source, Target, Proxy</p>	Cloud Foundry UAA Server
<code>uaa.origin.filter.enabled</code>	<p>This flag property depends on <code>uaa.origin</code>.</p> <p>If the flag is set to <i>true</i>, the Identity Provisioning service will read only users whose identity provider is set as a value of <code>uaa.origin</code>.</p> <p><b>Possible values:</b> <i>true</i> or <i>false</i></p> <ul style="list-style-type: none"> <li>• If set to <i>true</i>, the Identity Provisioning service will read only users whose identity provider is set as a value of <code>uaa.origin</code>.</li> <li>• If set to <i>false</i>, the Identity Provisioning service will read all users, regardless of their origin.</li> <li>• If set to <i>true</i> but the <code>uaa.origin</code> property is missing, the provisioning job will fail.</li> </ul> <p><b>System Role:</b> Source, Proxy</p>	Cloud Foundry UAA Server

Name	Description	System Type
<code>uaa.patch.response.with.re source</code>	<p>Use this property if you want to retrieve a group whose membership was modified.</p> <div> <p><b>Note</b></p> <p>This property is usable only when you have configured membership modifications via Add/Remove Member UAA endpoints. That is, when the <code>scim.support.patch.operation</code> property is set to <i>false</i>.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i> – the Identity Provisioning service will return the modified group via the GET /Groups endpoint of UAA. To learn how, see <a href="#">Retrieve</a>.</li> <li><i>false</i> – no modified groups will be returned by the service.</li> </ul> <p><b>System Role:</b> Proxy</p>	Cloud Foundry UAA Server

Name	Description	System Type
uaa.group.prefix	<p>This property distinguishes Cloud Foundry UAA Server groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>UAA_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the Cloud Foundry UAA Server source system and will be provisioned to the target system with the following name pattern: <b>UAA_&lt;GroupDisplayName&gt;</b>. This way Cloud Foundry UAA Server groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the Cloud Foundry UAA Server groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>UAA_</b> prefix in their display name will be provisioned to Cloud Foundry UAA Server. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to Cloud Foundry UAA Server.</li> </ul> <p><b>System Role:</b> Source and Target</p>	Cloud Foundry UAA Server

Name	Description	System Type
<code>xsuaa.origin</code>	<p>Holds the identity provider of a user in SAP BTP XS Advanced UAA (Cloud Foundry).</p> <p>You can find it in the SAP BTP cockpit. Go to your Cloud Foundry subaccount, choose <a href="#">Trust Configuration</a> and see the value under <a href="#">Origin Key</a>. For more information, see <a href="#">Configure Single and Multiple Origins [page 782]</a></p> <p><b>Possible values:</b> Text/numeric string</p> <p>For example: <a href="#">myaccount-xsuaa.accounts.ondemand.com</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP BTP XS Advanced UAA (Cloud Foundry)
<code>xsuaa.origin.filter.enabled</code>	<p>This flag property depends on <code>xsuaa.origin</code>.</p> <p>If the flag is set to <a href="#">true</a>, the Identity Provisioning service will read only users whose identity provider is set as a value of <code>xsuaa.origin</code>.</p> <p>Possible values: <a href="#">true</a> or <a href="#">false</a></p> <ul style="list-style-type: none"> <li>• If set to <a href="#">true</a>, the Identity Provisioning service will read only users whose identity provider is set as a value of <code>xsuaa.origin</code>.</li> <li>• If set to <a href="#">false</a>, the Identity Provisioning service will read all users, regardless of their origin.</li> <li>• If set to <a href="#">true</a> but the <code>xsuaa.origin</code> property is missing, the provisioning job will fail.</li> </ul> <p><b>System Role:</b> Source, Proxy</p>	SAP BTP XS Advanced UAA (Cloud Foundry)



Name	Description	System Type
<code>xsuaa.patch.response.with.resource</code>	<p>Use this property if you want to retrieve a group whose membership was modified.</p> <div> <p><b>Note</b></p> <p>This property is usable only when you have configured membership modifications via <a href="#">Add/Remove Member</a> UAA endpoints. That is, when the <code>scim.support.patch.operation</code> property is set to <code>false</code>.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> – the Identity Provisioning service will return the modified group via the GET <code>/Groups</code> endpoint of UAA. To learn how, see <a href="#">Retrieve</a>.</li> <li><code>false</code> – no modified groups will be returned by the service.</li> </ul> <p><b>System Role:</b> Proxy</p>	SAP BTP XS Advanced UAA (Cloud Foundry)
<code>workzone.user.filter</code>	<p>When specified, only those SAP Build Work Zone, advanced edition users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example: <code>userName eq "SmithJ"</code></p> <p><b>System Role:</b> Source</p>	SAP Build Work Zone, advanced edition
<code>workzone.group.filter</code>	<p>When specified, only those SAP Build Work Zone, advanced edition groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example: <code>displayName eq "ProjectTeam1"</code></p> <p><b>System Role:</b> Source</p>	SAP Build Work Zone, advanced edition

Name	Description	System Type
<code>workzone.content.type</code>	<p>This property makes a SAP Build Work Zone, advanced edition connector to send a specified value for the <i>Content-Type</i> HTTP header. This is needed because SAP Build Work Zone, advanced edition could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>Default value: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Build Work Zone, advanced edition

Name	Description	System Type
<code>workzone.support.patch.operation</code>	<p>The default value of this property is <i>false</i>. But for SAP Build Work Zone, advanced edition proxy systems, this property appears during creation and its predefined value is <i>true</i>. That means, when the Identity Provisioning identifies a changed entity in the back-end system, it will execute the updates as PATCH requests instead of PUT. That is, only changes will be written in SAP Build Work Zone, advanced edition, instead of provisioning the whole entity data.</p> <p><b>Additional Information:</b></p> <p>There are different cases when an entity should be updated in the target system:</p> <ul style="list-style-type: none"> <li>• In the source system, some of the entity attributes have been changed, or new attributes have been added.</li> <li>• In the source system, a condition or a filter is set for this entity not to be read anymore.</li> <li>• The whole entity has been deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <code>deleteEntity</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value for proxy systems: <i>true</i></p> <p>Default value for target systems: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Build Work Zone, advanced edition

Name	Description	System Type
<code>workzone.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists in SAP Build Work Zone, advanced edition. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s). This property defines by which unique attribute(s) the existing user to be searched (resolved).</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> <li>• <i>emails[0].value</i></li> <li>• <i>userName,emails[0].value</i></li> <li>• <i>externalId</i>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>Default value: <i>userName</i></p>	SAP Build Work Zone, advanced edition

Name	Description	System Type
<b>System Role:</b> Target, Proxy		
<code>workzone.group.unique.attribute</code>	<p>If the Identity Provisioning tries to create a group that already exists on the SAP Build Work Zone, advanced edition target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific). To make the search filter by a specific attribute, specify this attribute as a value for this property.</p> <p><b>Possible values:</b></p> <p>Default value (when not specified): <i>displayName</i></p> <p>If the property is not specified, the search is done by the default attribute: <code>displayName</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Build Work Zone, advanced edition
<code>fsm.group.filter</code>	<p>When specified, only those SAP Field Service Management groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example: <i>displayName eq "ProjectTeam1"</i></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Field Service Management
<code>fsm.user.filter</code>	<p>When specified, only those SAP Field Service Management users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example: <i>userName eq "SmithJ"</i></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Field Service Management


Name	Description	System Type
fsm.content.type	<p>This property makes the SAP Field Service Management connector to send a specified value for the <i>Content-Type</i> HTTP header. This is needed because SAP Field Service Management could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>Default value: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Field Service Management
fsm.group.unique.attribute	<p>If the Identity Provisioning tries to create a group that already exists in the SAP Field Service Management target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific). To make the search filter by a specific attribute, specify this attribute as a value for this property.</p> <p><b>Possible values:</b></p> <p>Default value (when not specified): <i>displayName</i></p> <p>If the property is not specified, the search is done by the default attribute: <i>displayName</i>.</p> <p><b>System Role:</b> Target, Proxy</p>	SAP Field Service Management

Name	Description	System Type
<code>fsm.include.if.match.wildcard.header</code>	<p>Makes the SAP Field Service Management connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by an SAP Field Service Management system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Field Service Management
<code>fsm.support.patch.operation</code>	<p>The default value of this property is <i>false</i>. But for SAP Field Service Management proxy systems, this property appears during creation and its predefined value is <i>true</i>. That means, when the Identity Provisioning identifies a changed entity in the back-end system, it will execute the updates as PATCH requests instead of PUT. That is, only changes will be written in SAP Field Service Management, instead of provisioning the whole entity data.</p> <p>Note that only attributes without "scope": "createEntity" in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set "scope": "createEntity".</p> <p><b>Possible values:</b></p> <p>Default value: <i>false</i></p> <p>Predefined value (during system creation): <i>true</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Field Service Management

Name	Description	System Type
fsm.user.unique.attribute	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists in SAP Field Service Management. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s). This property defines by which unique attribute(s) the existing user to be searched (resolved).</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> <li>• <i>emails[0].value</i></li> <li>• <i>userName,emails[0].value</i></li> </ul> <p>Default value: <i>userName</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Field Service Management



Name	Description	System Type
ias.api.version	<p>Defines the version of Identity Authentication SCIM API.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">1</a> - Identity Authentication SCIM API is used.</li> <li>• <a href="#">2</a> - Identity Directory SCIM API is used.</li> </ul> <p>Default value: <a href="#">2</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	Identity Authentication

Name	Description	System Type
ias.user.filter	<p>This property filters users by attributes from the SCIM core schema, the Enterprise user resource schema and the Custom defined schema. For example: <code>userName</code>, <code>emails.value</code>, <code>addresses.country</code>, <code>employeeNumber</code>, <code>costCenter</code>, <code>department</code> and others.</p> <p>For more information on the attributes defined in the SCIM core schema and the Enterprise user resource schema, see <a href="#">Identity Directory Service Schema View</a> </p> <p>You can set a single attribute or multiple ones as search criteria in the following value pattern:</p> <p>Single attribute: <code>&lt;user_attribute&gt; eq "&lt;value&gt;"</code></p> <p>Multiple attributes: <code>&lt;user_attribute1&gt; eq "&lt;value1&gt;" and/or &lt;user_attribute2&gt; eq "&lt;value2&gt;"</code></p> <p><b>Possible values:</b></p> <p>For example:</p> <ul style="list-style-type: none"> <li>Single attribute: <code>userName eq "Sebastian"</code></li> <li>Multiple attributes (with OR): <code>userName eq "Sebastian" or addresses.country eq "France"</code></li> <li>Multiple attributes (with AND): <code>userName eq "Sebastian" and addresses.country eq "France"</code></li> <li>Multiple attributes (with brackets): <code>userName eq "Sebastian" or (addresses.country eq "France" and emails.value eq "sebastian123@mail.com")</code></li> <li>Multiple attributes (enterprise attributes): <code>urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:department eq "Dev" and</code></li> </ul>	Identity Authentication (SCIM API version 2)

Name	Description	System Type
	<p><i>urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:organization eq "Technology"</i></p> <p><b>System Role:</b> Source, Proxy</p>	
<code>ias.group.filter</code>	<p>This property filters groups by display name.</p> <p>You can set a single display name or multiple ones as filter criteria. If you enter multiple display names (using OR operator), the filter will search for any of them.</p> <p>Single attribute: <i>displayName eq "&lt;group_name&gt;"</i></p> <p>Multiple attributes: <i>displayName eq "&lt;group_name1&gt;" or displayName eq "&lt;group_name2&gt;"</i></p> <p><b>Possible values:</b></p> <p>For example:</p> <ul style="list-style-type: none"> <li>• Single attribute: <i>displayName eq "FellowshipTeam1"</i></li> <li>• Multiple attributes: <i>displayName eq "FellowshipTeam1" or displayName eq "JuniorTest3"</i></li> </ul> <p><b>System Role:</b> Source, Proxy</p>	Identity Authentication (SCIM API version 2)

Name	Description	System Type
<code>ias.support.patch.operation</code>	<p>This property controls how modified entities (users and groups) in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, Identity Provisioning sends a <code>PATCH</code> request to the user or group resource in the target system. Only attributes without <code>"scope"</code> in the attribute mappings in the write transformation will be updated.</li> </ul> <p>For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with <code>"scope": "createEntity"</code>, such as:</p> <pre> {   "constant": true,   "targetPath": "\$[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ] [ 'mailVerified' ]",   "scope": "createEntity" }</pre> <ul style="list-style-type: none"> <li>If set to <i>false</i>, Identity Provisioning sends a <code>PUT</code> request to the user or group resource in the target system. This means, for example, that if a user attribute is modified or a group member is removed from a group, all user attributes and all group attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p>Users and groups can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>In the source system, some user or group attributes are modified, or new attributes are added.</li> </ul>	Identity Authentication (SCIM API version 2)

Name	Description	System Type
	<ul style="list-style-type: none"> <li>In the source system, a condition or a filter is set for users or groups not to be read anymore.</li> <li>A user or a group is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <code>deleteEntity</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	
<code>ias.user.groups.paging.enabled</code>	<p>This property enables paging of user's groups.</p> <p>The maximum number of user's groups returned per request is 1000. To read more than 1000 user's groups, paging must be enabled.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> - Paging is enabled. You can read more than 1000 user's groups in one request.</li> <li><code>false</code> - Paging is disabled. You can read up to 1000 user's groups in one request.</li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Source, Proxy</p>	Identity Authentication (SCIM API version 2)

Name	Description	System Type
<code>ias.group.members.paging.enabled</code>	<p>This property enables paging of group members.</p> <p>The maximum number of group members returned per request is 20 000. To read more than 20 000 group members, paging must be enabled.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> - Paging is enabled. You can read more than 20 000 group members in one request.</li> <li><code>false</code> - Paging is disabled. You can read up to 20 000 group members in one request.</li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Source, Proxy</p>	Identity Authentication (SCIM API version 2)
<code>ias.include.if.match.wildcard.header</code>	<p>Makes the connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by a SCIM system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	Identity Authentication (SCIM API version 2)

Name	Description	System Type
<code>ias.user.unique.attribute</code>	<p>When Identity Provisioning attempts to provision a user for the first time, it may detect that this user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user will be searched and resolved. <b>If the service finds a user on the target system via this filter, then the conflicting user will overwrite the existing one.</b> If the service does not find a user on the target system via this filter, the creation will fail.</p> <p>According to your use case and system type, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is set during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>phoneNumbers[0].value</i>. If the service finds an existing user with such <i>phoneNumber</i>, it</li> </ul>	Identity Authentication (SCIM API version 2)

Name	Description	System Type
	<p>updates this user with the data of the conflicting one. If a user with such <i>phoneNumber</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</p> <ul style="list-style-type: none"> <li>Value = <i>userName</i>, <i>phoneNumbers[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>phoneNumber</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>Value = <i>userName</i>, <i>emails[0].value</i>, <i>phoneNumbers[0].value</i>. If the service finds an existing user with these <i>userName</i>, <i>email</i> and <i>phoneNumber</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>userName</i></li> <li><i>emails[0].value</i></li> <li><i>userName,emails[0].value</i></li> <li><i>phoneNumbers[0].value</i></li> <li><i>userName</i>, <i>phoneNumbers[0].value</i></li> <li><i>userName</i>, <i>emails[0].value</i>, <i>phoneNumbers[0].value</i></li> <li><i>externalId</i>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>Default value: <i>userName</i></p> <p><b>System Role:</b> Target, Proxy</p>	



Name	Description	System Type
<code>ias.group.unique.attribute</code>	<p>If you try to provision a group that already exists in a target system, the group creation will fail. In this case, the existing group only needs to be updated.</p> <p>This property defines by which unique attribute(s) the existing group will be searched and resolved. The default value is <i>displayName</i>. Currently, it is the only unique attribute that is supported. When set, you can expect the following behavior:</p> <ul style="list-style-type: none"> <li>• If a group with the given <i>displayName</i> is found in the target system, the group that you try to provision will overwrite the existing one.</li> <li>• If a group with the given <i>displayName</i> is not found in the target system, the group that you try to provision will not be created in the target system.</li> </ul> <p><b>Possible values:</b></p> <p>If the property is not specified, the search is done by the default attribute: <code>displayName</code></p> <p><b>System Role:</b> Target, Proxy</p>	Identity Authentication (SCIM API version 2)

Name	Description	System Type
<code>ias.content.type</code>	<p>Makes the connector send a specified value for the <i>Content-Type</i> HTTP header. This is needed because a SCIM system could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>If the property is not specified, the default value is taken: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	Identity Authentication (SCIM API version 2)

Name	Description	System Type
<code>aad.user.filter.group.filter.combine</code>	<p>Filters Microsoft Azure AD users based on their group assignments.</p> <p>When set to <i>true</i>, this property combines user and group filters defined on the <code>aad.user.filter</code> and <code>aad.group.filter</code> properties to further narrow the search results. This way, only users that meet the following filtering criteria are returned:</p> <ul style="list-style-type: none"> <li>• Users that match the user filter and at the same time are members of groups that match the group filter.</li> <li>• Members of the filtered groups that match the user filter.</li> </ul> <div> <p><b>Note</b></p> <p>To make the <code>aad.user.filter.group.filter.combine</code> property work, ensure that both user and group entities are read, that is, neither of them is ignored in the transformation code.</p> </div> <div> <p><b>❖ Example</b></p> <ol style="list-style-type: none"> <li>1. You have the following users, located in two cities with one or more assigned groups:  <b>User:</b> David Thompson from London with assigned <b>Groups:</b> Marketing and Sales  <b>User:</b> Julie Armstrong from New York with assigned <b>Groups:</b> Employee  <b>User:</b> John Smith from New York with assigned <b>Groups:</b> Marketing and Sales</li> <li>2. You have defined the following filtering criteria:  <code>aad.user.filter = city eq "New York"</code></li> </ol> </div>	Microsoft Azure Active Directory

Name	Description	System Type
	<div> <pre>aad.group.filter = displayName eq "Marketing" aad.user.filter.group.filter.combine = true</pre> <p>3. You get the following result: Only user John Smith is returned as it matches the user filter and at the same time is a member of the group that matches the group filter. Although David Thompson matches the group filter, he doesn't match the user filter. Although Julie Armstrong matches the user filter, she doesn't match the group filter.</p> <p>When set to <i>false</i>, user and group filters are not combined.</p> <p>For more information, see: <a href="#">Identity Provisioning: How to Get Users Based on Group Assignments from MS Azure AD</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i> When this property is set to <b>true</b>, it is expected that filtering criteria are defined for <code>aad.user.filter</code> and <code>aad.group.filter</code> properties. If one or both are not defined in the UI, be aware of the following behavior: <ul style="list-style-type: none"> <li>Only <code>aad.user.filter</code> is defined: Users that match the user filter and are members of any group will be returned. If a user matches the user filter but is not a member of any group, this user will not be returned.</li> <li>Only <code>aad.group.filter</code> is defined: Users that are</li> </ul> </li> </ul> </div>	

Name	Description	System Type
	<p>members of the groups matching the group filter will be returned.</p> <ul style="list-style-type: none"> <li>None of the properties are defined: Users that are members of any group will be returned.</li> </ul> <p>When this property is set to <i>true</i> and filtering criteria are defined for <code>aad.user.filter</code> and <code>aad.group.filter</code> properties, if you are searching for a specific user or group using Identity Provisioning service API, be aware of the following behavior:</p> <ul style="list-style-type: none"> <li>When searching for specific user with GET <code>.../Users/UserId</code> request, filtering criteria defined on both properties are not considered. The user is returned with all the groups he or she is a member of.</li> <li>When searching for specific group with GET <code>.../Groups/GroupId</code> request, filtering criteria defined on both properties are not considered. The group is returned with all its group members.</li> <li><i>false</i> - default value</li> </ul> <p><b>System Role:</b> Source, Proxy</p>	
<code>s4hana.pp.user.filter</code>	<p>When specified, only those SAP S/4HANA for procurement planning users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>Example: <i>name.familyName eq "Smith" and addresses.country eq "US"</i></p> <p><b>System Role:</b> Source, Proxy</p>	SAP S/4HANA for procurement planning

Name	Description	System Type
<code>s4hana.pp.content.type</code>	<p>Makes the connector send a specified value for the <i>Content-Type</i> HTTP header. This is needed because a SCIM system could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>If the property is not specified, the default value is taken: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP S/4HANA for procurement planning
<code>s4hana.pp.include.if.match.wildcard.header</code>	<p>Makes the connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by a SCIM system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP S/4HANA for procurement planning

Name	Description	System Type
s4hana.pp.user.unique.attribute	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). <b>If the service finds such a user on the target system via this filter, then the conflicting user will overwrite the existing one.</b> If the service does not find such a user, the creation will fail.</p> <p>According to your use case and system type, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> </ul>	SAP S/4HANA for procurement planning

Name	Description	System Type
	<ul style="list-style-type: none"> <li><code>emails[0].value</code></li> <li><code>userName,emails[0].value</code></li> <li><code>externalId</code>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>Default value: <code>userName</code></p> <p><b>System Role:</b> Target</p>	



Name	Description	System Type
<code>ias.user.automatic.conflict.resolution</code>	<p>Controls whether automatic conflict resolution is switched on or off in Identity Authentication (target system) when provisioning is triggered from source systems containing different users with the same user identifiers (IDs).</p> <p>For example, when SAP SuccessFactors and SAP SuccessFactors Learning are configured as source systems for provisioning users to Identity Authentication, it could happen that different SAP SuccessFactors and SAP SuccessFactors Learning users have the same user IDs. In this case, when the first user is created in Identity Authentication, after triggering a provisioning job, the second (conflicting) user will either overwrite the already existing one (automatic conflict resolution is switched on) or will fail and won't be created (automatic conflict resolution is switched off).</p> <p>To control this behavior, you can use the <code>ias.user.automatic.conflict.resolution</code> property in the target Identity Authentication system. This property is not added by default.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> - If the property is set to true, or is not set at all, the automatic conflict resolution is switched on. This means that Identity Provisioning takes into account the unique attribute(s) defined on the <code>scim.user.unique.attribute</code> property (when using SCIM API version 1) or <code>ias.user.unique.attribute</code> property (when using SCIM API version 2) and tries to find an already existing user in Identity</li> </ul>	Identity Authentication

Name	Description	System Type
	<p>Authentication matching these attributes.</p> <ul style="list-style-type: none"> <li>If a user is found, the provisioning of a new (conflicting) user is resolved as follows: the conflicting user overwrites the existing one.</li> <li>If a user is not found, the provisioning of a conflicting user fails, and it is not created in Identity Authentication.</li> <li><i>false</i> - If the property is set to false, the automatic conflict resolution is switched off. This means that Identity Provisioning does not take into account the unique attribute(s) defined on the <code>scim.user.unique.attribute</code> property (when using SCIM API version 1) or <code>ias.user.unique.attribute</code> property (when using SCIM API version 2) and fails the provisioning of a conflicting user. This user is not created in Identity Authentication and does not overwrite the existing one. In the Job Log, an error code 409 , uniqueness will be displayed.</li> </ul> <p>Default value: <i>true</i></p> <p><b>System Role:</b> Target</p>	

Name	Description	System Type
<code>ias.patch.group.members.above.threshold</code>	<p>Defines the threshold number of group members above which they are provisioned on batches with PATCH requests, and below which they are provisioned with PUT request. Setting this property allows you to avoid timeouts when updating groups with a large number of group members.</p> <div> <p><b>Note</b></p> <p>You can use this property when Identity Authentication is based on Identity Directory SCIM API (in short, SCIM API version 2).</p> </div> <p><b>Possible values:</b> integer</p> <p>Default value: <i>20 000</i></p> <p>Minimum value: <i>1</i></p> <p>Maximum value: <i>20 000</i></p> <p>For example:</p> <ul style="list-style-type: none"> <li><b>PATCH requests:</b> If you have a group with 700 members and you update the group by adding another 1200 members, setting this property to 900 results in the following: As 1900 (the target count of the members) is above the threshold number of 900, 2 PATCH requests will be sent to the Identity Authentication target system. The first request will add 900 group members and the second request will add 300 group members. The threshold number you set defines the maximum number of group members processed per batch.</li> <li><b>PUT request:</b> If you have a group with 700 members and you update the group by adding another 100</li> </ul>	Identity Authentication

Name	Description	System Type
	<p>members, setting this property to 900 results in the following:</p> <p>As 800 (the target count of the members) is below the threshold number of 900, 1 PUT request with 800 group members will be sent to the Identity Authentication target system to update the group.</p> <div data-bbox="603 658 992 987"> <p><b>i Note</b></p> <p>Regardless of the threshold number you define, when removing group members in Identity Authentication, the maximum number of members which can be removed per one PATCH request is 50.</p> </div> <p><b>System Role:</b> Target</p>	

Name	Description	System Type
<code>idds.patch.group.members.above.threshold</code>	<p>Defines the threshold number of group members above which they are provisioned on batches with PATCH requests, and below which they are provisioned with PUT request. Setting this property allows you to avoid timeouts when updating groups with a large number of group members.</p> <div> <p><b>Note</b></p> <p>You can use this property when Identity Authentication and Identity Provisioning (where Local Identity Directory is configured), are running on the same infrastructure, that is, the infrastructure of Identity Authentication.</p> </div> <p><b>Possible values:</b> integer</p> <p>Default value: <i>20 000</i></p> <p>Minimum value: <i>1</i></p> <p>Maximum value: <i>200 000</i></p> <p>For example:</p> <ul style="list-style-type: none"> <li><b>PATCH requests:</b> If you have a group with 700 members and you update the group by adding another 1200 members, setting this property to 900 results in the following: As 1900 (the target count of the members) is above the threshold number of 900, 2 PATCH requests will be sent to the Local Identity Directory target system. The first request will add 900 group members and the second request will add 300 group members. The threshold number you set defines the maximum number of group members processed per batch.</li> </ul>	Local Identity Directory

Name	Description	System Type
	<ul style="list-style-type: none"> <li>PUT request: If you have a group with 700 members and you update the group by adding another 100 members, setting this property to 900 results in the following: As 800 (the target count of the members) is below the threshold number of 900, 1 PUT request with 800 group members will be sent to the Local Identity Directory target system to update the group.</li> </ul> <div> <p><b>i Note</b></p> <p>Regardless of the threshold number you define, when removing group members in Local Identity Directory, the maximum number of members which can be removed per one PATCH request is 50.</p> </div>	
	<b>System Role:</b> Target	

Name	Description	System Type
<code>xsuaa.patch.group.members.above.threshold</code>	<p>Defines the threshold number of group members above which they are provisioned on batches with <code>PATCH</code> requests, and below which they are provisioned with <code>PUT</code> request. Setting this property allows you to avoid timeouts when updating groups with a large number of group members.</p> <p><b>Possible values:</b> integer</p> <p>Default value: <code>20 000</code></p> <p>Minimum value: <code>1</code></p> <p>Maximum value: <code>200 000</code></p> <p>For example:</p> <ul style="list-style-type: none"> <li> <b><code>PATCH</code> requests:</b> If you have a group with 700 members and you update the group by adding another 1200 members, setting this property to 900 results in the following:  As 1900 (the target count of the members) is above the threshold number of 900, 2 <code>PATCH</code> requests will be sent to the XSUAA target system. The first request will add 900 group members and the second request will add 300 group members.  The threshold number you set defines the maximum number of group members processed per batch. </li> <li> <b><code>PUT</code> request:</b> If you have a group with 700 members and you update the group by adding another 100 members, setting this property to 900 results in the following:  As 800 (the target count of the members) is below the threshold number of 900, 1 <code>PUT</code> request with 800 group members will be sent to the XSUAA target system to update the group. </li> </ul> <p><b>System Role:</b> Target</p>	SAP BTP XS Advanced UAA (Cloud Foundry)

Name	Description	System Type
<code>ias.user.update.instead.delete</code>	<p>When using SCIM API version 2, this property allows you to update user attributes with <code>PATCH</code> request in Identity Authentication target system and to preserve the user record instead of deleting it. This behavior is supported only when the scope of the attribute is set to <b>deleteEntity</b>.</p> <p>In addition to configuring this property, you also need to adapt the write transformation. For example, if you want to disable a user account in Identity Authentication, you need to do the following:</p> <ol style="list-style-type: none"> <li>1. Set <code>ias.user.update.instead.delete=true</code></li> <li>2. Adapt the write transformation as follows:</li> </ol> <pre> {   "user": {     "mappings": [       {         "constant": "urn:ietf:params:scim:api:messages:2.0:PatchOp",         "targetPath": "\$.schemas[0]",         "scope": "deleteEntity"       },       {         "constant": "replace",         "targetPath": "\$.Operations[0].op",         "scope": "deleteEntity"       },       {         "constant": "active",         "targetPath": "\$.Operations[0].path",         "scope": "deleteEntity"       }     ]   } } </pre>	Identity Authentication




Name	Description	System Type
	<pre>       },       {         "constant": false,         "targetPath": "\$.Operations[0].value",         "scope": "deleteEntity"       },       ... </pre> <p>In this case, the PATCH operation will replace <i>true</i> with <i>false</i> as the value of the active user attribute. As a result, when the PATCH operation is executed, the user record in the target system will no longer be managed by Identity Provisioning as it is considered deleted.</p> <p>For more information, see: <a href="#">Transformation Expressions [page 330]</a> → <a href="#">Scope</a> → <a href="#">deleteEntity</a> → <a href="#">Identity Authentication (SCIM API version 2)</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p>When the property is set to <i>true</i>, adapt the write transformation with the attribute name and the attribute value you want to update:</p> <pre> {   "user": {     "mappings": [       {         "constant": "urn:ietf:params:scim:api:messages:2.0:PatchOp",         "targetPath": "\$.schemas[0]",         "scope": "deleteEntity"       },       {         "constant": "replace", </pre>	

Name	Description	System Type
	<pre> "targetPath": "\$.Operations[0].op", "scope": "deleteEntity"     },     { "constant": "&lt;attribute_name&gt;", "targetPath": "\$.Operations[0].path", "scope": "deleteEntity"     },     { "constant": "&lt;attribute_value&gt;", "targetPath": "\$.Operations[0].value", "scope": "deleteEntity"     },     ... </pre>	
	<b>System Role:</b> Target	
lms.user.filter	<p>When specified, only those users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>userName eq "testName"</i></li> <li><i>externalID eq "testID"</i></li> <li><i>active eq "true"</i></li> <li><i>sourceSystem eq "Learning"</i> - indicates that the user is created directly in SAP SuccessFactors Learning with no involvement of Identity Provisioning.</li> <li><i>sourceSystem eq "Identity Provisioning"</i> - indicates that the user is created in SAP SuccessFactors Learning by Identity Provisioning.</li> </ul> <p><b>System Role:</b> Source</p>	SAP SuccessFactors Learning

Name	Description	System Type
<code>lms.content.type</code>	<p>This property makes the SAP SuccessFactors Learning connector to send a specified value for the <i>Content-Type</i> HTTP header. This is needed because SAP SuccessFactors Learning could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>Default value: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP SuccessFactors Learning
<code>lms.user.unique.attribute</code>	<p>When Identity Provisioning attempts to provision a user for the first time, it may detect that such user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). If the service finds such user on the target system via this filter, then the conflicting user will overwrite the existing one. If the service does not find such a user, the creation will fail.</p> <p><b>Default behavior:</b> The property is missing during system creation. Its default value is <i>userName</i>. This means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such <i>userName</i> is not found, the creation of the conflicting user fails.</p> <p><b>Possible values:</b></p> <p>Default value: <i>userName</i></p> <p><b>System Role:</b> Target</p>	SAP SuccessFactors Learning

Name	Description	System Type
<code>lms.include.if.match.wildcard.header</code>	<p>Makes the connector send the <a href="#">If-Match</a> HTTP header with a value of "*" for every request to the target system. This header could be used by a SCIM system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">true</a></li> <li><a href="#">false</a></li> </ul> <p><b>System Role:</b> Target, Proxy</p>	SAP SuccessFactors Learning

Name	Description	System Type
<code>lms.support.patch.operation</code>	<p>This property controls how modified users in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, Identity Provisioning sends a PATCH request to the user or group resource in the target system. Only attributes without "scope": "createEntity" in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set "scope": "createEntity".</li> <li>If set to <i>false</i>, PUT operations are used to update users in the target system. This means, for example, that if a user attribute is modified, all user attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p>Users can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>In the source system, some user attributes are modified, or new attributes are added.</li> <li>In the source system, a condition or a filter is set for users not to be read anymore.</li> <li>A user is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <b>deleteEntity</b>.</p>	SAP SuccessFactors Learning

Name	Description	System Type
	<p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">true</a></li> <li><a href="#">false</a></li> </ul> <p>Default value: <a href="#">false</a></p> <p><b>System Role:</b> Target</p>	
<code>lms.instance.host</code>	<p>Enter the host of your SAP SuccessFactors Learning instance.</p> <p>This property must be configured if you want to use client certificate authentication for the communication between Identity Provisioning and SAP SuccessFactors Learning.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP SuccessFactors Learning
<code>ias.support.bulk.operation</code>	<p>This property enables bulk operations for users and groups.</p> <p>When bulk operations are enabled, Identity Provisioning creates, updates, and deletes multiple users and groups in one request.</p> <p>When bulk operations are not enabled, Identity Provisioning creates, updates, and deletes one user at a time.</p> <p>For more information, see: <a href="#">Identity Directory SCIM API</a> .</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">true</a> - bulk operations are enabled</li> <li><a href="#">false</a> - bulk operations are not enabled</li> </ul> <p>Default value: <a href="#">false</a></p> <p><b>System Role:</b> Target</p>	Identity Authentication (using SCIM API version 2)

Name	Description	System Type
<code>ias.bulk.operations.max.count</code>	<p>This property sets the number of operations to be performed in one bulk request.</p> <p><b>Possible values:</b></p> <p>Default value: <i>20</i></p> <p>Maximum value: <i>100</i></p> <p>If you enter a number larger than 100, Identity Provisioning will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	Identity Authentication (using SCIM API version 2)
<code>ids.support.bulk.operation</code>	<p>This property enables bulk operations for users and groups.</p> <p>When bulk operations are enabled, Identity Provisioning creates, updates, and deletes multiple users and groups in one request.</p> <p>When bulk operations are not enabled, Identity Provisioning creates, updates, and deletes one user at a time.</p> <p>For more information, see: <a href="#">Identity Directory SCIM API</a>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i> - bulk operations are enabled</li> <li><i>false</i> - bulk operations are not enabled</li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	Local Identity Directory (when Identity Provisioning is running on SAP Cloud Identity Infrastructure)
<code>ids.bulk.operations.max.count</code>	<p>This property sets the number of operations to be performed in one bulk request.</p> <p><b>Possible values:</b></p> <p>Default value: <i>20</i></p> <p>Maximum value: <i>100</i></p> <p>If you enter a number larger than 100, Identity Provisioning will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	Local Identity Directory (when Identity Provisioning is running on SAP Cloud Identity Infrastructure)

Name	Description	System Type
<code>concur.user.filter</code>	<p>When specified, only those users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example:</p> <ul style="list-style-type: none"> <li> <a href="#"><code>userName eq "johnsmith@example.com"</code></a>  As the userName must be unique across SAP Concur, this filter returns only the user matching this userName. </li> <li> <a href="#"><code>companyId eq "aa067ada-71a9-4f57-8e98-9300b1c3171d"</code></a>  This filter returns all users in the company with this companyId. </li> <li> <a href="#"><code>externalId eq "0fe44868-31a7-4930-9ah30-757tg2513b64"</code></a>  This filter returns a user with the specified value, that is, the user-UUID generated for the user in Identity Authentication. </li> <li> <a href="#"><code>employeeNumber eq "Concur Administrator"</code></a>  This filter returns a user with the specified employee number. The employeeNumber could also be a number having six or more digits. </li> </ul>	SAP Concur (using SAP Concur Identity v4 API)
<b>System Role:</b> Source		



Name	Description	System Type
<code>concur.content.type</code>	<p>Makes the connector send a specified value for the <i>Content-Type</i> HTTP header. This is needed because a SCIM system could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>If the property is not specified, the default value is taken: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Concur (using SAP Concur Identity v4 API)
<code>concur.include.if.match.wildcard.header</code>	<p>Makes the connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by a SCIM system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Concur (using SAP Concur Identity v4 API)

Name	Description	System Type
<code>concur.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). <b>If the service finds such a user on the target system via this filter, then the conflicting user will overwrite the existing one.</b> If the service does not find such a user, the creation will fail.</p> <p>According to your use case and system type, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> </ul>	SAP Concur (using SAP Concur Identity v4 API)

Name	Description	System Type
	<ul style="list-style-type: none"> <li><code>emails[0].value</code></li> <li><code>userName,emails[0].value</code></li> <li><code>externalId</code>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>Default value: <code>userName</code></p> <p><b>System Role:</b> Target</p>	
<code>concur.datacenter</code>	<p>The SAP Concur data center your Identity Provisioning tenant belongs to.</p> <p>Based on the provided data center, Identity Provisioning configures the URL of the User Provisioning Service (UPS) v4 API or the SAP Concur Identity v4 API.</p> <p>For example, if you provide <code>us1</code>, the service will configure the URL in the following pattern: <code>us.api.concursolutions.com</code>.</p> <p><b>Possible values:</b></p> <p>The following SAP Concur data centers are available:</p> <ul style="list-style-type: none"> <li><code>us1</code></li> <li><code>us2</code></li> <li><code>eu1</code></li> <li><code>eu2</code></li> <li><code>emea</code></li> <li><code>cn1</code></li> <li><code>usg</code></li> <li><code>int</code></li> </ul> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Concur
<code>concur.api.version</code>	<p>Defines the version of SAP Concur API.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>1</code> - SAP Concur User Provisioning Service (UPS) v4 API is used.</li> <li><code>2</code> - SAP Concur Identity v4 API (SCIM API) is used.</li> </ul> <p>Default value: <code>2</code></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Concur

Name	Description	System Type
<code>concur.authorization.code</code>	<p>(Credential)</p> <p>Enter the <a href="#">Company Request Token</a> and run a provisioning job within 24 hours from generating the token in the SAP Concur Company Request Token self-service tool. Otherwise, the token will expire, and you'll need a new one.</p> <p>After the first run of the job, Identity Provisioning fills in automatically a refresh token as the value of the <code>concur.refresh.token</code> property. If a provisioning job has not been run for six months, you'll again need to generate a new token.</p> <div> <p>→ Remember</p> <p>The company request token has a 24 hour validity. If this token expires, you must request a new token.</p> <p>The refresh token has a six month validity. Every time you run a provisioning job, the validity of the refresh token is extended with six months starting from the date of the last run. If you haven't run a provisioning job for six months, your refresh token will expire and you must request a new company request token.</p> </div> <p>The Company Request Token is generated in the SAP Concur Company Request Token self-service tool.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Concur
<code>concur.company.id</code>	<p>Your company UUID</p> <p>The Company ID is generated in the SAP Concur Company Request Token self-service tool.</p> <p><b>System Role:</b> Target, Proxy</p>	SAP Concur

Name	Description	System Type
<code>concur.company.domain</code>	<p>Your company domain</p> <p>The username and the company domain are concatenated in the SAP Concur default transformations in the following format: <code>user@domain</code></p> <p>Your company domain is the part of your username behind the @ symbol. For example: <code>johnsmith@example.com</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Concur
<code>ariba.applications.patch.group.members.of.nested.groups</code>	<p>If you set this property to <code>true</code>, Identity Provisioning will update only user members of a group in SAP Ariba Applications target system. The update will be executed on batches via PATCH requests. This will preserve the group hierarchy with nested groups in the SAP Ariba Applications backend.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target</p>	SAP Ariba Applications
<code>ariba.applications.patch.group.members.above.threshold</code>	<p>This property is relevant only when <code>ariba.applications.patch.group.members.of.nested.groups</code> is set to <code>true</code>.</p> <p>It defines the maximum number of user members of a group that are included in one PATCH request. If the maximum value of 200 000 is exceeded, the system sets automatically the default value.</p> <p><b>Possible values:</b> integer</p> <p>Default value: <code>20 000</code></p> <p>Minimum value: <code>1</code></p> <p>Maximum value: <code>200 000</code></p> <p><b>System Role:</b> Target</p>	SAP Ariba Applications

Name	Description	System Type
<code>cflp.providerId</code>	<p>Your SAP Build Work Zone, standard edition provider ID</p> <p>The provider ID is specified in the Channel Manager of the SAP Build Work Zone, standard edition when defining a new content provider. For more information about configuring the content provider to use the Identity Provisioning service, see <a href="#">Configure Integration with the Identity Provisioning Service</a></p> <p><b>Possible values:</b></p> <p>The value of your SAP Build Work Zone, standard edition provider ID</p> <p>For example: <a href="#">ABC123</a></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Build Work Zone, standard edition
<code>cflp.user.filter</code>	<p>When specified, only those SAP Build Work Zone, standard edition users matching the filter expression will be read.</p> <p>By default, users are always filtered by the <a href="#">providerId</a>. If another filtering attribute is defined, for example the email of the user, both filters are combined.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">emails.value eq 'john.smith@example.com'</a></li> </ul> <div data-bbox="665 1440 756 1473" data-label="Section-Header"> <h4>i Note</h4> </div> <div data-bbox="665 1494 973 1628" data-label="Text"> <p>Although, the email is supported as a filtering attribute, it is not returned when searching for the user.</p> </div> <ul style="list-style-type: none"> <li><a href="#">urn:ietf:params:scim:schemas:extension:2.0:mapping.providerId eq 'ABC123'</a></li> </ul> <p><b>System Role:</b> Proxy</p>	SAP Build Work Zone, standard edition

Name	Description	System Type
cflp.group.filter	<p>When specified, only those SAP Build Work Zone, standard edition groups matching the filter expression will be read. By default, groups are always filtered by the <i>providerId</i>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>externalId eq 12345678</i></li> <li><i>urn:ietf:params:scim:schemas:extension:2.0:mapping:providerId eq 'ABC123'</i></li> <li><i>meta.isIAG eq true</i></li> </ul> <p>This filtering attribute indicates whether the group will be used in a hybrid scenario with SAP Cloud Identity Access Governance.</p> <p><b>System Role:</b> Proxy</p>	SAP Build Work Zone, standard edition

Name	Description	System Type
<code>cflp.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p><b>Possible values:</b></p> <p>SAP Build Work Zone, standard edition supports the following unique attributes which are automatically filled in when the target system is added in the service UI:</p> <pre>emails[0].value, [ 'urn:ietf:params:scim:schemas:extension:2.0:mapping' ] [ 'providerId' ], externalId</pre> <ul style="list-style-type: none"> <li>• If the user has an <b>externalId</b>, the conflict is resolved by <b>externalId</b> and <b>providerId</b>.</li> <li>• If the user doesn't have an <b>externalId</b>, the conflict is resolved by <b>email</b> and <b>providerId</b>.</li> </ul> <p>For the conflict to be resolved, an existing user matching both unique attributes should be found. If an existing user doesn't match both unique attributes or matches only one of them, the user creation fails.</p> <div> <p>→ Recommendation</p> <p>We recommend that you do not modify the value of the <code>cflp.user.unique.attribute</code> property. Otherwise, user creation fails.</p> </div>	SAP Build Work Zone, standard edition
<b>System Role:</b> Target, Proxy		



Name	Description	System Type
<code>cflp.group.unique.attribute</code>	<p>If Identity Provisioning tries to provision a group that already exists in the target system (a conflicting group), this property defines the unique attributes by which the existing group will be searched and resolved.</p> <p><b>Possible values:</b></p> <p>SAP Build Work Zone, standard edition supports a pair of unique attributes which is automatically filled in when the target system is added in the service UI:</p> <p><i><code>externalId</code>,  <code>['urn:ietf:params:scim:schemas:extension:2.0:mapping']['providerId']</code></i></p> <p>For the conflict to be resolved, an existing group matching both unique attributes should be found. In this case, Identity Provisioning updates the group. This means, the conflicting group overwrites the existing one. If the group matches only one of the unique attributes, the conflict is not resolved, and the group creation fails.</p> <div> <p>→ Recommendation</p> <p>We recommend that you do not modify the value of the <code>cflp.group.unique.attribute</code> property. Otherwise, the group creation fails.</p> </div>	SAP Build Work Zone, standard edition
<b>System Role:</b> Target, Proxy		

Name	Description	System Type
<code>cflp.patch.group.members.above.threshold</code>	<p>Defines the threshold number of group members above which they are provisioned on batches with PATCH requests, and below which they are provisioned with PUT request. Setting this property allows you to avoid timeouts when updating groups with a large number of group members.</p> <p><b>Possible values:</b> integer</p> <p>Default and maximum value: <a href="#">5000</a></p> <p>Minimum value: <a href="#">1</a></p> <p>For example:</p> <ul style="list-style-type: none"> <li> <p>PATCH requests: If you have a group with 700 members and you update the group by adding another 1200 members, setting this property to 900 results in the following:</p> <p>As 1900 (the target count of the members) is above the threshold number of 900, 2 PATCH requests will be sent to the SAP Build Work Zone, standard edition target system. The first request will add 900 group members and the second request will add 300 group members.</p> <p>The threshold number you set defines the maximum number of group members processed per batch.</p> </li> <li> <p>PUT request: If you have a group with 700 members and you update the group by adding another 100 members, setting this property to 900 results in the following:</p> <p>As 800 (the target count of the members) is below the threshold number of 900, 1 PUT request with 800 group members will be sent to the SAP Build Work Zone, standard edition target system to update the group.</p> </li> </ul>	SAP Build Work Zone, standard edition

Name	Description	System Type
	<div> <b>i Note</b> <p>If the maximum value of 5 000 is exceeded, the system will automatically use the default value.</p> </div>	
	<b>System Role:</b> Target	
cflp.support.bulk.operation	<p>This property enables bulk operations for users and groups.</p> <p>When the property is enabled (set to <i>true</i>), the following operations can be executed by Identity Provisioning service in a single request:</p> <ul style="list-style-type: none"> <li>• Create or delete multiple users</li> <li>• Create, update, or delete multiple groups</li> </ul> <p>When the property is disabled (set to <i>false</i>), the following operations can be executed by Identity Provisioning service for a single entity at a time:</p> <ul style="list-style-type: none"> <li>• Create or delete a user</li> <li>• Create, update, or delete a group</li> </ul> <div> <b>i Note</b> <p>Update operation is skipped for users in the default write transformation.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP Build Work Zone, standard edition

Name	Description	System Type
cflp.bulk.operations.max.count	<p>This property sets the number of operations to be performed in one bulk request.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">20</a></p> <p>Maximum value: <a href="#">100</a></p> <p>If you enter a number larger than 100, the service will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	SAP Build Work Zone, standard edition

Name	Description	System Type
<code>xsuaa.user.unique.attribute</code>	<p>When Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). If the service finds such a user on the target system via this filter, then the conflicting user will overwrite the existing one. If the service does not find such a user, the creation will fail.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both attributes: <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul>	SAP BTP XS Advanced UAA (Cloud Foundry)

Name	Description	System Type
	<ul style="list-style-type: none"> <li>Value = <i>userName,origin</i>. If the service finds an existing user with both attributes: <i>userName</i> and <i>origin</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>userName</i></li> <li><i>emails[0].value</i></li> <li><i>userName,emails[0].value</i></li> <li><i>userName,origin</i></li> <li><i>externalId</i>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>Default value: <i>userName</i></p> <p><b>System Role:</b> Target</p>	

Name	Description	System Type
fsm.group.prefix	<p>This property distinguishes SAP Field Service Management groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>FSM_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Field Service Management source system and will be provisioned to the target system with the following name pattern: <b>FSM_&lt;GroupDisplayName&gt;</b>. This way SAP Field Service Management groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Field Service Management groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>FSM_</b> prefix in their display name will be provisioned to SAP Field Service Management. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Field Service Management.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP Field Service Management

Name	Description	System Type
<code>sf.user.read.deactivatedafter</code>	<p>This property filters SAP SuccessFactors inactive users from a particular date on. It is an optional property which does not appear by default at system creation. It accepts a value in the <b>yyyy-MM-dd</b> format. For example: <b>2023-07-17</b></p> <p>The <code>sf.user.read.deactivatedafter</code> property is supported for SAP SuccessFactors version 2 using SAP SuccessFactors Workforce SCIM API. It works together with the <code>sf.user.filter</code> property which is added at system creation with the default value: <i>active eq true</i>. Using it can further narrow down the filtering results.</p> <p>To filter active users along with inactive ones from a particular date on, the following configuration must be in place:</p> <ul style="list-style-type: none"> <li>Set the <code>sf.user.read.deactivatedafter</code> value to a date in the expected format. For example: <i>2023-07-17</i></li> <li><code>sf.user.filter = active eq true</code></li> </ul> <p>As a result, Identity Provisioning reads SAP SuccessFactors active users and the users set to inactive from that date on using the 2023-07-17T00:00:00Z date-time format.</p> <p>Depending on the value you define for <code>sf.user.filter</code>, expect the following results:</p> <ul style="list-style-type: none"> <li><code>sf.user.filter = active eq false</code> All inactive users will be returned.</li> <li><code>sf.user.filter = active eq false and userName sw "Test_"</code></li> </ul>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)



Name	Description	System Type
	<p>All inactive users with username starting with Test_ will be returned.</p> <div> <p><b>i Note</b></p> <p>When you filter by <code>sf.user.filter = active eq false</code> along with the property <code>sf.user.read.deactivatedafter</code>, the users that match the two criteria will be read twice.</p> </div> <ul style="list-style-type: none"> <li>• <code>sf.user.filter = active eq true and userName sw "Test_"</code> Inactive users from the provided date on and all active users with username starting with Test_ will be returned.</li> </ul> <p><b>System Role:</b> Source, Proxy</p>	

Name	Description	System Type
<code>cc.user.filter</code>	<p>When specified, only those users matching the filter expression will be read. You can filter users by <b><code>userName</code></b>, <b><code>emails.value</code></b>, and <b><code>externalId</code></b>, according to the API syntax of SAP Commerce Cloud.</p> <p><b>Possible values:</b> text/ numeric string</p> <p>For example:</p> <ul style="list-style-type: none"> <li><code>userName eq "johnbrown" and externalId eq "P000252"</code></li> <li><code>userName eq "johnbrown" and emails.value eq "johnbrown@email.com"</code></li> <li><code>userName eq "johnbrown" and emails.value eq "johnbrown@email.com" and externalId eq "P000252"</code></li> </ul> <div> <p><b>i Note</b></p> <p>These combinations are valid for both 'or' and 'and' operators.</p> </div> <p><b>System Role:</b> Source, Proxy</p>	SAP Commerce Cloud
<code>cc.group.filter</code>	<p>When specified, only those groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p>For example:</p> <p><code>displayName eq "ProjectTeam1" or "Students2018"</code></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Commerce Cloud

Name	Description	System Type
<code>cc.support.patch.operation</code>	<p>This property controls how modified entities (users and groups) in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, Identity Provisioning sends a <code>PATCH</code> request to the user or group resource in the target system. Only attributes without <code>"scope": "createEntity"</code> in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set <code>"scope": "createEntity"</code>.</li> <li>If set to <i>false</i>, <code>PUT</code> operations are used to update users and groups in the target system. This means, for example, that if a user attribute is modified or a group member is removed from a group, all user attributes and all group attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <div> <p><b>Note</b></p> <p>When executing a remove operation, all members of a group are removed without considering the number of deleted group members in the source system.</p> </div> <p><b>Additional Information:</b></p> <p>There are different cases when an entity should be updated in the target system:</p> <ul style="list-style-type: none"> <li>In the source system, some of the entity attributes have been changed, or new attributes have been added.</li> </ul>	SAP Commerce Cloud

Name	Description	System Type
	<ul style="list-style-type: none"> <li>In the source system, a condition or a filter is set for this entity not to be read anymore.</li> <li>The whole entity has been deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <code>deleteEntity</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value for proxy systems: <code>true</code></p> <p>Default value for target systems: <code>false</code></p> <p><b>System Role:</b> Proxy, Target</p>	
<code>cc.include.if.match.wildcard.header</code>	<p>Makes the SAP Commerce Cloud connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by an SAP Commerce Cloud system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Commerce Cloud

Name	Description	System Type
cc.group.unique.attribute	<p>If you try to provision a group that already exists in a target system, the group creation will fail. In this case, the existing group only needs to be updated.</p> <p>This property defines by which unique attribute(s) the existing group will be searched and resolved. The default value is <i>displayName</i>. Currently, it is the only unique attribute that is supported. When set, you can expect the following behavior:</p> <ul style="list-style-type: none"> <li>• If a group with the given <i>displayName</i> is found in the target system, the group that you try to provision will overwrite the existing one.</li> <li>• If a group with the given <i>displayName</i> is not found in the target system, the group that you try to provision will not be created in the target system.</li> </ul> <p><b>Possible values:</b></p> <p>If the property is not specified, the search is done by the default attribute: <i>displayName</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Commerce Cloud

Name	Description	System Type
<code>cc.user.unique.attribute</code>	<p>When Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). If the service finds such a user on the target system via this filter, then the conflicting user will overwrite the existing one. If the service does not find such a user, the creation will fail.</p> <p>The property is automatically added during system creation. If the service finds an existing user by at least one of the uniqueness criteria, which are <i>email</i>, <i>userName</i>, or <i>externalId</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the update of the conflicting user fails. If more than one users with these unique attributes are found, the update fails.</p> <p><b>Possible values:</b> <i>emails[0].value</i>, <i>userName</i>, <i>externalId</i></p> <p>Default value: <i>emails[0].value</i>, <i>userName</i>, <i>externalId</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Commerce Cloud



Name	Description	System Type
<code>cc.content.type</code>	<p>Makes the connector send a specified value for the <i>Content-Type</i> HTTP header. This is needed because a SCIM system could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>If the property is not specified, the default value is taken: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Commerce Cloud

Name	Description	System Type
<code>cc.patch.group.members.above.threshold</code>	<p>Defines the threshold number of group members above which they are provisioned on batches with PATCH requests, and below which they are provisioned with PUT request. Setting this property allows you to avoid timeouts when updating groups with a large number of group members.</p> <div> <p><b>i Note</b></p> <p>You can use this property when SAP Commerce Cloud is based on SAP Commerce Cloud SCIM API (in short, SCIM API version 2).</p> </div> <p><b>Possible values:</b> integer</p> <p>Default value: <i>20 000</i></p> <p>Minimum value: <i>1</i></p> <p>Maximum value: <i>200 000</i></p> <p>For example:</p> <ul style="list-style-type: none"> <li><b>PATCH requests:</b> If you have a group with 700 members and you update the group by adding another 1200 members, setting this property to 900 results in the following: As 1900 (the target count of the members) is above the threshold number of 900, 2 PATCH requests will be sent to the Identity Authentication target system. The first request will add 900 group members and the second request will add 300 group members. The threshold number you set defines the maximum number of group members processed per batch.</li> <li><b>PUT request:</b> If you have a group with 700 members and you update the group by adding another 100</li> </ul>	SAP Commerce Cloud



Name	Description	System Type
	<p>members, setting this property to 900 results in the following:</p> <p>As 800 (the target count of the members) is below the threshold number of 900, 1 PUT request with 800 group members will be sent to the Identity Authentication target system to update the group.</p> <div> <p><b>i Note</b></p> <p>Regardless of the threshold number you define, when removing group members in SAP Commerce Cloud, the maximum number of members which can be removed per one PATCH request is 98.</p> </div> <p><b>System Role:</b> Target</p>	
<code>cc.patch.group.members.of.nested.groups</code>	<p>If you set this property to <i>true</i>, Identity Provisioning will update only user members of a group in SAP Commerce Cloud target system. The update will be executed on batches via PATCH requests. This will preserve the group hierarchy with nested groups in the SAP Commerce Cloud backend.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP Commerce Cloud

Name	Description	System Type
<code>maco.roles.prefix</code>	<p>This property distinguishes SAP Market Communication for Utilities roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SMC_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the SAP Market Communication for Utilities source system and will be provisioned to the target system with the following name pattern: <b>SMC_&lt;role_name&gt;</b>. This way SAP Market Communication for Utilities roles in the target system will be distinguished from roles provisioned from other applications.</li> </ul> <p>If the property is not set, the SAP Market Communication for Utilities roles will be read and provisioned to the target system with their actual role names.</p> <ul style="list-style-type: none"> <li>When <b>set in the target system</b>, only roles containing the <b>SMC_</b> prefix in their role name will be provisioned to SAP Market Communication for Utilities. Roles without this prefix in the role name won't be provisioned.</li> </ul> <p>If the property is not set, all roles will be provisioned to SAP Market Communication for Utilities.</p> <p><b>System Role:</b> Source and Target</p>	SAP Market Communication for Utilities

Name	Description	System Type
<code>maco.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP Market Communication for Utilities target or proxy system.</p> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li><code>false</code> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>Default value (if the property is missing during system creation): <code>true</code></p> <p>Default value (if the property appears during system creation): <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Market Communication for Utilities
<code>maco.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP Market Communication for Utilities system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a>  → 4.5 Filter System Query Option</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Market Communication for Utilities

Name	Description	System Type
<code>maco.roles.page.size</code>	<p>This property indicates how many business roles (considered as <i>groups</i>) per page to be read from your SAP Market Communication for Utilities source system.</p> <p><b>Possible values:</b> Integer number</p> <p>For example, if you set the property's value = <i>30</i>, the Identity Provisioning will read 30 roles (groups) at once, then – another 30, and so on.</p> <p><b>System Role:</b> Source, Proxy</p>	SAP Market Communication for Utilities
<code>maco.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source For example, if you set the property's value = In the event of archived (disabled) entities in a source nullnullnullSAP Market Communication for Utilities</p> <p>In the source systems, this property is activated by default. If you want to always read disabled entities, set the property to <i>false</i>, or delete it.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>true</i></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Market Communication for Utilities
<code>maco.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request.</p> <p>For more information, see: <a href="#">APIs for Business User Management</a></p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP Market Communication for Utilities

Name	Description	System Type
<code>maco.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">20</a></p> <p>Maximum value: <a href="#">100</a></p> <p>If you enter a number larger than 100, the service will replace it with the default value (20).</p> <p><b>System Role:</b> Target</p>	SAP Market Communication for Utilities
<code>sf.api.version</code>	<p>Handles the version of the API which is consumed by the SAP SuccessFactors system.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">1</a> - SAP SuccessFactors HCM Suite OData API (in short, OData API) is used.</li> <li><a href="#">2</a> - SAP SuccessFactors Workforce SCIM API (in short, SCIM API) is used.</li> </ul> <p>Default value: <a href="#">1</a></p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>SAP SuccessFactors (using version 1 - SAP SuccessFactors HCM Suite OData API)</li> <li>SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)</li> </ul>
<code>sf.company.id</code>	<p>Enter the Company ID of your SAP SuccessFactors system.</p> <p>The Company ID is a short string of characters that identifies each SAP SuccessFactors system. It is like a username for your organization. All users of the same system share the same Company ID.</p> <p>This property must be configured if you want to use client certificate authentication for the communication between Identity Provisioning and SAP SuccessFactors.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	<ul style="list-style-type: none"> <li>SAP SuccessFactors (using version 1 - SAP SuccessFactors HCM Suite OData API)</li> <li>SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)</li> </ul>

Name	Description	System Type
<code>sf.support.patch.operation</code>	<p>This property controls how modified users in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, Identity Provisioning sends a PATCH request to the user or group resource in the target system. Only attributes without "scope": "createEntity" in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set "scope": "createEntity".</li> <li>If set to <i>false</i>, PUT operations are used to update users in the target system. This means, for example, that if a user attribute is modified, all user attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <div> <p><b>Note</b></p> <p>Updating the user attributes: <code>name</code> and <code>manager</code>, is not supported with PATCH operation.</p> </div> <p>Users can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>In the source system, some user attributes are modified, or new attributes are added.</li> <li>In the source system, a condition or a filter is set for users not to be read anymore.</li> <li>A user is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code></p>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)

Name	Description	System Type
	<p>scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <b>deleteEntity</b>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	
<code>sf.include.if.match.wildcard.header</code>	<p>Makes the connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by a SCIM system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p><b>System Role:</b> Target, Proxy</p>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)
<code>sf.group.unique.attribute</code>	<p>If the service tries to create a group that already exists in the target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific).</p> <p>To make the search filter by a specific attribute, specify this attribute as a value for the <code>sf.group.unique.attribute</code> property.</p> <p>If the property is not specified, the search is done by the default attribute: <i>displayName</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)

Name	Description	System Type
<code>sf.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). <b>If the service finds such a user on the target system via this filter, then the conflicting user will overwrite the existing one.</b> If the service does not find such a user, the creation will fail.</p> <p>According to your use case and system type, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> </ul>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)



Name	Description	System Type
	<ul style="list-style-type: none"> <li><code>emails[0].value</code></li> <li><code>userName,emails[0].value</code></li> <li><code>externalId</code>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>Default value: <code>userName</code></p> <p><b>System Role:</b> Target</p>	
<code>sf.content.type</code>	<p>This property makes the SAP SuccessFactors connector to send a specified value for the <code>Content-Type</code> HTTP header. This is needed because SAP SuccessFactors could potentially not implement the protocol in the specification, which states that a system must accept <code>application/scim+json</code> as a value of the <code>Content-Type</code> header.</p> <p><b>Possible values:</b></p> <p>For example: <code>application/json</code></p> <p>Default value: <code>application/scim+json</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP SuccessFactors (using version 1 - SAP SuccessFactors HCM Suite OData API)

Name	Description	System Type
<code>sf.patch.group.members.above.threshold</code>	<p>Defines the threshold number of group members above which they are provisioned on batches with <code>PATCH</code> requests, and below which they are provisioned with <code>PUT</code> request. Setting this property allows you to avoid timeouts when updating groups with a large number of group members.</p> <div> <p><b>Note</b></p> <p>You can use this property when SAP SuccessFactors is based on SAP SuccessFactors Workforce SCIM API (in short, SCIM API).</p> </div> <p><b>Possible values:</b> integer</p> <p>Default value: <i>20 000</i></p> <p>Minimum value: <i>1</i></p> <p>Maximum value: <i>200 000</i></p> <p>For example:</p> <ul style="list-style-type: none"> <li><b><code>PATCH</code> requests:</b> If you have a group with 700 members and you update the group by adding another 1200 members, setting this property to 900 results in the following: As 1900 (the target count of the members) is above the threshold number of 900, 2 <code>PATCH</code> requests will be sent to the Local Identity Directory target system. The first request will add 900 group members and the second request will add 300 group members. The threshold number you set defines the maximum number of group members processed per batch.</li> <li><b><code>PUT</code> request:</b> If you have a group with 700 members and you update the group by adding another 100</li> </ul>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)

Name	Description	System Type
	<p>members, setting this property to 900 results in the following:</p> <p>As 800 (the target count of the members) is below the threshold number of 900, 1 PUT request with 800 group members will be sent to the Local Identity Directory target system to update the group.</p> <p><b>System Role:</b> Target</p>	
<code>sf.group.members.paging.enabled</code>	<p>This property enables paging of group members.</p> <p>The maximum number of group members returned per request is 100. To read more than 100 group members, paging must be enabled.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code> - Paging is enabled.</li> <li><code>false</code> - Paging is disabled.</li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Source, Proxy</p>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)
<code>sf.patch.group.members.of.nested.groups</code>	<p>If you set this property to <code>true</code>, Identity Provisioning will update only user members of a group in SAP SuccessFactors target system. The update will be executed on batches via PATCH requests. This will preserve the group hierarchy with nested groups in the SAP SuccessFactors backend.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target</p>	SAP SuccessFactors (using version 2 - SAP SuccessFactors Workforce SCIM API)

Name	Description	System Type
sac.group.prefix	<p>This property distinguishes SAP Analytics Cloud groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SAC_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Analytics Cloud source system and will be provisioned to the target system with the following name pattern: <b>SAC_&lt;GroupDisplayName&gt;</b>. This way SAP Analytics Cloud groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Analytics Cloud groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>SAC_</b> prefix in their display name will be provisioned to SAP Analytics Cloud. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Analytics Cloud.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP Analytics Cloud

Name	Description	System Type
cpq.group.prefix	<p>This property distinguishes SAP CPQ groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>CPQ_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP CPQ source system and will be provisioned to the target system with the following name pattern: <b>CPQ_&lt;GroupDisplayName&gt;</b>. This way SAP CPQ groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP CPQ groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>CPQ_</b> prefix in their display name will be provisioned to SAP CPQ. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP CPQ.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP CPQ
cpq.user.filter	<p>When specified, only those SAP CPQ users matching the filter expression will be read.</p> <p>Example: <b>name.familyName eq "Smith" and addresses.country eq "US"</b></p> <p><b>System Role:</b> Source, Proxy</p>	SAP CPQ

Name	Description	System Type
scp.group.prefix	<p>This property distinguishes SAP BTP Account Members (Neo) groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SCP_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP BTP Account Members (Neo) source system and will be provisioned to the target system with the following name pattern: <b>SCP_&lt;GroupDisplayName&gt;</b>. This way SAP BTP Account Members (Neo) groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP BTP Account Members (Neo) groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>SCP_</b> prefix in their display name will be provisioned to SAP BTP Account Members (Neo). Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP BTP Account Members (Neo).</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP BTP Account Members (Neo)

Name	Description	System Type
<code>c4c.user.filter</code>	<p>When specified, only those SAP Sales Cloud and SAP Service Cloud users matching the filter expression will be read.</p> <p>SAP Sales Cloud and SAP Service Cloud is formerly known as SAP Cloud for Customer (in short, C4C).</p> <p>For example:</p> <ul style="list-style-type: none"> <li>• <code>userName eq "Smith"</code></li> <li>• <code>email eq "test@abc.com"</code></li> <li>• <code>employeeNumber eq "56789"</code></li> <li>• <code>addresses.country eq "USA"</code></li> </ul> <p><b>System Role:</b> Source, Proxy</p>	SAP Sales Cloud and SAP Service Cloud
<code>c4c.group.filter</code>	<p>When specified, only those SAP Sales Cloud and SAP Service Cloud groups matching the filter expression will be read.</p> <p>SAP Sales Cloud and SAP Service Cloud is formerly known as SAP Cloud for Customer (in short, C4C).</p> <p>Example: <b><code>displayName eq "Project-Team1"</code></b></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Sales Cloud and SAP Service Cloud

Name	Description	System Type
<code>xsuaa.group.prefix</code>	<p>This property distinguishes SAP BTP XS Advanced UAA (Cloud Foundry) groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>XSUAA_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP BTP XS Advanced UAA (Cloud Foundry) source system and will be provisioned to the target system with the following name pattern: <b>XSUAA_&lt;GroupDisplayName&gt;</b>. This way SAP BTP XS Advanced UAA (Cloud Foundry) groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP BTP XS Advanced UAA (Cloud Foundry) groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>XSUAA_</b> prefix in their display name will be provisioned to SAP BTP XS Advanced UAA (Cloud Foundry). Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP BTP XS Advanced UAA (Cloud Foundry).</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP BTP XS Advanced UAA (Cloud Foundry)



Name	Description	System Type
<code>cflp.support.patch.operation</code>	<p>This property controls how modified entities (users and groups) in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, Identity Provisioning sends a <code>PATCH</code> request to the user or group resource in the target system. Only attributes without <code>"scope": "createEntity"</code> in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set <code>"scope": "createEntity"</code>.</li> <li>If set to <i>false</i>, <code>PUT</code> operations are used to update users and groups in the target system. This means, for example, that if a user attribute is modified or a group member is removed from a group, all user attributes and all group attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <div> <p><b>Note</b></p> <p>Updating users and groups returns 404 Not Found.</p> </div> <p>Users and groups can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>In the source system, some user or group attributes are modified, or new attributes are added.</li> <li>In the source system, a condition or a filter is set for users or groups not to be read anymore.</li> <li>A user or a group is deleted from the source system.</li> </ul>	SAP Build Work Zone, standard edition

Name	Description	System Type
	<p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <code>deleteEntity</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <code>true</code></li> <li>• <code>false</code></li> </ul> <p>Default value for proxy systems: <code>true</code></p> <p>Default value for target systems: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	
<code>sac.api.version</code>	<p>Defines the version of SAP Analytics Cloud SCIM API.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <code>1</code> - SAP Analytics Cloud SCIM API version 1 is used.</li> <li>• <code>2</code> - SAP Analytics Cloud SCIM API version 2 is used.</li> </ul> <p>Default value: <code>1</code></p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Analytics Cloud

Name	Description	System Type
<code>sac.support.patch.operation</code>	<p>This property controls how modified entities (users and groups) in the source system are updated in the target system.</p> <div> <p><b>i Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, PATCH operations are used to update users and groups in the target system. This means, for example, that if a user attribute is modified or a group member is removed from a group, only these changes will be provisioned and applied in the target system.</li> <li>If set to <i>false</i>, PUT operations are used to update users and groups in the target system. This means, for example, that if a user attribute is modified or a group member is removed from a group, all user attributes and all group attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p> </div>	SAP Analytics Cloud

Name	Description	System Type
<code>sac.include.if.match.wildcard.header</code>	<p>This property makes the SAP Analytics Cloud connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. The header could be used by an SAP Analytics Cloud system for entity versioning.</p> <div> <p><b>Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Analytics Cloud
<code>sac.group.unique.attribute</code>	<p>If the Identity Provisioning tries to create a group that already exists in the SAP Analytics Cloud target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific). To make the search filter by a specific attribute, specify this attribute as a value for this property.</p> <div> <p><b>Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> </div> <p><b>Possible values:</b></p> <p>Default value: <i>displayName</i></p> <p>If the property is not specified, the search is done by the default attribute: <i>displayName</i>.</p> <p><b>System Role:</b> Target, Proxy</p>	SAP Analytics Cloud

Name	Description	System Type
<code>sac.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists in SAP Analytics Cloud. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s). This property defines by which unique attribute(s) the existing user to be searched (resolved).</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <div> <p><b>Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> <li>• <i>emails[0].value</i></li> <li>• <i>userName,emails[0].value</i></li> </ul>	SAP Analytics Cloud

Name	Description	System Type
	<p>Default value: <a href="#">userName</a></p> <p><b>System Role:</b> Target, Proxy</p>	
sac.content.type	<p>Makes the connector send a specified value for the <a href="#">Content-Type</a> HTTP header. This is needed because a SCIM system could potentially not implement the protocol in the specification, which states that a system must accept <a href="#">application/scim+json</a> as a value of the <a href="#">Content-Type</a> header.</p> <div> <p><b>i Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> </div> <p><b>Possible values:</b></p> <p>For example: <a href="#">application/json</a></p> <p>If the property is not specified, the default value is taken: <a href="#">application/scim+json</a></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Analytics Cloud
sac.user.filter	<p>When specified, only those SAP Analytics Cloud users matching the filter expression will be read.</p> <div> <p><b>i Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> </div> <p><b>Possible values:</b></p> <p>For example: <a href="#">userName eq "SmithJ"</a></p> <p><b>System Role:</b> Source</p>	SAP Analytics Cloud

Name	Description	System Type
sac.group.filter	<p>When specified, only those SAP Analytics Cloud groups matching the filter expression will be read.</p> <div> <p><b>i Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> </div> <p><b>Possible values:</b></p> <p>For example: <i>displayName eq "ProjectTeam1"</i></p> <p><b>System Role:</b> Source</p>	SAP Analytics Cloud

Name	Description	System Type
<code>sac.patch.group.members.above.threshold</code>	<p>Defines the threshold number of group members above which they are provisioned on batches with PATCH requests, and below which they are provisioned with PUT request. Setting this property allows you to avoid timeouts when updating groups with a large number of group members.</p> <div> <p><b>i Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> </div> <p><b>Possible values:</b> integer</p> <p>Default value: <i>20 000</i></p> <p>Minimum value: <i>1</i></p> <p>Maximum value: <i>200 000</i></p> <p>For example:</p> <ul style="list-style-type: none"> <li> <b>PATCH requests:</b> If you have a group with 700 members and you update the group by adding another 1200 members, setting this property to 900 results in the following:  As 1900 (the target count of the members) is above the threshold number of 900, 2 PATCH requests will be sent to the SAP Analytics Cloud target system. The first request will add 900 group members and the second request will add 300 group members.  The threshold number you set defines the maximum number of group members processed per batch. </li> <li> <b>PUT request:</b> If you have a group with 700 members and you update the group by adding another 100 </li> </ul>	SAP Analytics Cloud



Name	Description	System Type
	<p>members, setting this property to 900 results in the following:</p> <p>As 800 (the target count of the members) is below the threshold number of 900, 1 PUT request with 800 group members will be sent to the SAP Analytics Cloud target system to update the group.</p> <div> <p><b>i Note</b></p> <p>Regardless of the threshold number you define, when removing group members in SAP Analytics Cloud, the maximum number of members which can be removed per one PATCH request is 98.</p> </div> <p><b>System Role:</b> Target</p>	
<code>sac.patch.group.members.of.nested.groups</code>	<p>If you set this property to <i>true</i>, Identity Provisioning will update only user members of a group in SAP Analytics Cloud target system. The update will be executed on batches via PATCH requests. This will preserve the group hierarchy with nested groups in the SAP Analytics Cloud backend.</p> <div> <p><b>i Note</b></p> <p>You can use this property when SAP Analytics Cloud is based on SCIM API version 2.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target</p>	SAP Analytics Cloud
<code>ep.user.filter</code>	<p>When specified, only those SAP Enterprise Portal users matching the filter expression will be read. For more information, see <a href="#">Filtering</a>.</p>	SAP Enterprise Portal

Name	Description	System Type
<code>ep.group.filter</code>	When specified, only those SAP Enterprise Portal groups matching the filter expression will be read. For more information, see <a href="#">Filtering</a> .	SAP Enterprise Portal
<code>bn.user.filter</code>	<p>When specified, only those SAP Business Network users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>userName eq "Julie Armstrong"</code></li> <li><code>userName sw "J"</code></li> <li><code>name.familyName eq "Armstrong"</code></li> <li><code>emails eq "julie.armstrong@example.com"</code></li> </ul> <p><b>System Role:</b> Source</p>	SAP Business Network
<code>bn.group.filter</code>	<p>When specified, only those SAP Business Network groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <p><code>displayName eq "Employees"</code></p> <p><b>System Role:</b> Source</p>	SAP Business Network
<code>bn.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved. The property is not added automatically at system creation.</p> <p>Default value: <code>userName</code></p> <p>If the service finds an existing user by <code>userName</code>, it updates this user with the data of the conflicting one. If the service does not find an existing user by <code>userName</code>, the creation of the conflicting user fails.</p> <p><b>System Role:</b> Target, Proxy</p>	SAP Business Network

Name	Description	System Type
<code>bn.include.if.match.wildcard.header</code>	<p>Makes the connector send the If-Match HTTP header with a value of "*" for every request to the target system. This header could be used by a SCIM system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Business Network

Name	Description	System Type
<code>bn.support.patch.operation</code>	<p>This property controls how modified entities (users and groups) in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, Identity Provisioning sends a <code>PATCH</code> request to the user or group resource in the target system. Only attributes without <code>"scope": "createEntity"</code> in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set <code>"scope": "createEntity"</code>.</li> <li>If set to <i>false</i>, <code>PUT</code> operations are used to update users and groups in the target system. This means, for example, that if a user attribute is modified or a group member is removed from a group, all user attributes and all group attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <div data-bbox="620 1375 975 1628"> <p><b>Note</b></p> <p>Updating a group by removing all its members doesn't work. When executing a remove operation, groups must have at least one member.</p> </div> <p>Users and groups can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>In the source system, some user or group attributes are modified, or new attributes are added.</li> <li>In the source system, a condition or a filter is set for users or groups not to be read anymore.</li> </ul>	SAP Business Network

Name	Description	System Type
	<ul style="list-style-type: none"> <li>A user or a group is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <code>deleteEntity</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value for proxy systems: <code>true</code></p> <p>Default value for target systems: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	
<code>bn.content.type</code>	<p>This property makes SAP Business Network connector to send a specified value for the <code>Content-Type</code> HTTP header. This is needed because SAP Business Network could potentially not implement the protocol in the specification, which states that a system must accept <code>application/scim+json</code> as a value of the <code>Content-Type</code> header.</p> <p><b>Possible values:</b></p> <p>For example: <code>application/json</code></p> <p>Default value: <code>application/scim+json</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Business Network
<code>bn.api.key</code>	<p>An API Key represents the unique key that identifies a particular application as a legitimate consumer of an API.</p> <p><b>System Role:</b>Source, Target, Proxy</p>	SAP Business Network
<code>bn.realm.id</code>	<p>The realm name is part of the URL you use to access SAP Business Network.</p> <p><b>System Role:</b>Source, Target, Proxy</p>	SAP Business Network

Name	Description	System Type
<code>abap.host.timezone</code>	<p>Specifies the time zone of SAP Application Server ABAP on-premise systems. The value is used for calculating the correct assignments validity in case your SAP AS ABAP and Identity Provisioning tenant are running in different time zones.</p> <p><b>Possible values:</b></p> <p>The value should be provided in the following format: <b>UTC+/- offset</b>. For example: <b>UTC+02:00</b>, <b>UTC-04:00</b>, <b>UTC+03:30</b>.</p> <p>Internet Assigned Numbers Authority (IANA) Time Zone database format is also supported. For more information, see <a href="#">RFC 6557: Procedures for Maintaining the Time Zone Database</a> .</p> <p><b>System Role:</b> Source</p>	SAP Application Server ABAP
<code>s4hana.afc.group.filter</code>	<p>When specified, only those SAP Advanced Financial Closing users matching the filter expression will be read.</p> <p>Supported operators: <b>eq</b> (equal), <b>sw</b> (starts with) and <b>co</b> (contains)</p> <p><b>Possible values:</b></p> <p>For example: <code>displayName eq "Administrators"</code></p> <p><b>System Role:</b> Source, Proxy</p>	SAP Advanced Financial Closing

Name	Description	System Type
<code>s4hana.afc.user.filter</code>	<p>When specified, only those SAP Advanced Financial Closing users matching the filter expression will be read.</p> <p>Supported operators: <b>eq</b> (equal), <b>sw</b> (starts with) and <b>co</b> (contains)</p> <p><b>Possible values:</b></p> <p>For example:</p> <ul style="list-style-type: none"> <li><code>userName eq "Julie Armstrong"</code></li> <li><code>emails eq "julie.armstrong@example.com"</code></li> <li><code>name.familyName sw "A"</code></li> <li><code>name.givenName co "Ju"</code></li> </ul> <p><b>System Role:</b> Source, Proxy</p>	SAP Advanced Financial Closing
<code>s4hana.afc.content.type</code>	<p>This property makes a SAP Advanced Financial Closing connector to send a specified value for the <i>Content-Type</i> HTTP header. This is needed because SAP Advanced Financial Closing could potentially not implement the protocol in the specification, which states that a system must accept <i>application/scim+json</i> as a value of the <i>Content-Type</i> header.</p> <p><b>Possible values:</b></p> <p>For example: <i>application/json</i></p> <p>Default value: <i>application/scim+json</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Advanced Financial Closing

Name	Description	System Type
<code>s4hana.afc.group.unique.tribute</code>	<p>If the Identity Provisioning tries to create a group that already exists on the SAP Advanced Financial Closing target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific). To make the search filter by a specific attribute, specify this attribute as a value for this property.</p> <p><b>Possible values:</b></p> <p>Default value: <i>displayName</i></p> <p>If the property is not specified, the search is done by the default attribute: <i>displayName</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Advanced Financial Closing
<code>s4hana.afc.include.if.match.wildcard.header</code>	<p>Makes the SAP Advanced Financial Closing connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by an SAP Advanced Financial Closing system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i></li> <li><i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Advanced Financial Closing



Name	Description	System Type
s4hana.afc.support.patch.operation	<p>This property controls how modified entities (users and groups) in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>• If set to <i>true</i>, PATCH operations are used to update users and groups in the target system. This means, for example, that if a user attribute is modified or a group member is removed from a group, only these changes will be provisioned and applied in the target system.</li> <li>• If set to <i>false</i>, PUT operations are used to update users and groups in the target system. This means, for example, that if a user attribute is modified or a group member is removed from a group, all user attributes and all group attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Advanced Financial Closing

Name	Description	System Type
s4hana.afc.user.unique.attribute	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Advanced Financial Closing target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property does not appear in the UI during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> <li>• <i>emails[0].value</i></li> <li>• <i>userName,emails[0].value</i></li> </ul> <p>Default value: <i>userName</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Advanced Financial Closing

Name	Description	System Type
s4hana.afc.group.prefix	<p>This property distinguishes SAP Advanced Financial Closing groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>AFC_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Advanced Financial Closing source system and will be provisioned to the target system with the following name pattern: <b>AFC_&lt;GroupDisplayName&gt;</b>. This way SAP Advanced Financial Closing groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Advanced Financial Closing groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>AFC_</b> prefix in their display name will be provisioned to SAP Advanced Financial Closing. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Advanced Financial Closing.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP Advanced Financial Closing

Name	Description	System Type
s4hana.pp.group.prefix	<div> <div><b>i Note</b></div> <div> <p>SAP S/4HANA for procurement planning does not support groups. Therefore, configuring the <code>s4hana.pp.group.prefix</code> property is irrelevant.</p> </div> </div> <p>This property distinguishes the groups of a given provisioning system by specific prefix that you provide. It is optional and does not appear by default at system creation.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix is prepended to the name of the groups and they are provisioned to the target system with the following name pattern: <b>&lt;YOUR_PREFIX&gt;_&lt;GroupDisplayName&gt;</b>. This way those groups are distinguished from groups provisioned from other applications. If the property is not set, the groups are read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing <b>YOUR_PREFIX</b> prefix in their display name are provisioned there. Groups without this prefix in the display name are not provisioned. If the property is not set, all groups are provisioned to the target system.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP S/4HANA for procurement planning

Name	Description	System Type
<code>s4hana.cloud.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP S/4HANA Cloud target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <div> <p><b>Note</b></p> <p>You can configure this property only when the integration between a human resource (HR) system and SAP S/4HANA Cloud target system is <b>OFF</b>. Since the HR integration is active and cannot be switched off for SAP S/4HANA Cloud target systems, configuring the <code>s4hana.cloud.user.unique.attribute</code> property is currently irrelevant.</p> </div> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>• Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is</li> </ul>	SAP S/4HANA Cloud

Name	Description	System Type
	<p>due to another reason, so the creation of the conflicting user fails.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <i>personExternalID</i></li> <li>• <i>emails[0].value</i></li> </ul> <p>Default value: <i>personExternalID</i></p> <p><b>System Role:</b> Target, Proxy</p>	

Name	Description	System Type
s4hana.onprem.user.unique.attribute	<p>If Identity Provisioning tries to provision a user that already exists in the SAP S/4HANA On-Premise target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property does not appear in the UI during system creation. Its default value is <i>personExternalID</i>. That means, if the service finds an existing user by a <i>personExternalID</i>, it updates this user with the data of the conflicting one. If a user with such a <i>personExternalID</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user matching both unique attributes <i>email</i> and <i>personExternalID</i>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <i>email</i>, the update of the existing user fails. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <i>personExternalID</i></li> <li>• <i>emails[0].value</i></li> </ul> <p>Default value: <i>personExternalID</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP S/4HANA On-Premise

Name	Description	System Type
<code>ibp.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Integrated Business Planning for Supply Chain target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>personExternalID</code></li> <li><code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Integrated Business Planning for Supply Chain



Name	Description	System Type
<code>maco.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Market Communication for Utilities target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>• Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <code>personExternalID</code></li> <li>• <code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Market Communication for Utilities

Name	Description	System Type
<code>marketing.cloud.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Marketing Cloud target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>personExternalID</code></li> <li><code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Marketing Cloud

Name	Description	System Type
a4c.user.unique.attribute	<p>If Identity Provisioning tries to provision a user that already exists in the SAP BTP ABAP environment target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property does not appear in the UI during system creation. Its default value is <i>personExternalID</i>. That means, if the service finds an existing user by a <i>personExternalID</i>, it updates this user with the data of the conflicting one. If a user with such a <i>personExternalID</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user matching both unique attributes <i>email</i> and <i>personExternalID</i>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <i>email</i>, the update of the existing user fails. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <i>personExternalID</i></li> <li>• <i>emails[0].value</i></li> </ul> <p>Default value: <i>personExternalID</i></p> <p><b>System Role:</b> Target, Proxy</p>	SAP BTP ABAP environment

Name	Description	System Type
c4c.group.prefix	<p>This property distinguishes SAP Sales Cloud and SAP Service Cloud groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>C4C_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Sales Cloud and SAP Service Cloud source system and will be provisioned to the target system with the following name pattern: <b>C4C_&lt;GroupDisplayName&gt;</b>. This way SAP Sales Cloud and SAP Service Cloud groups in the target system will be distinguished from groups provisioned from other applications.</li> <li>When <b>set in the target system</b>, only groups containing the <b>C4C_</b> prefix in their display name will be provisioned to SAP Sales Cloud and SAP Service Cloud. Groups without this prefix in the display name won't be provisioned.</li> </ul> <p>If the property is not set, the SAP Sales Cloud and SAP Service Cloud groups will be read and provisioned to the target system with their actual display names.</p> <p><b>System Role:</b> Source, Target</p>	SAP Sales Cloud and SAP Service Cloud

Name	Description	System Type
s4hana.cloud.roles.prefix	<p>This property distinguishes SAP S/4HANA Cloud roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>S4HANA_CLOUD_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the SAP S/4HANA Cloud source system and will be provisioned to the target system with the following name pattern: <b>S4HANA_CLOUD_&lt;role_name&gt;</b> . This way SAP S/4HANA Cloud roles in the target system will be distinguished from roles provisioned from other applications. If the property is not set, the SAP S/4HANA Cloud roles will be read and provisioned to the target system with their actual role names.</li> <li>When <b>set in the target system</b>, only roles containing the <b>S4HANA_CLOUD_</b> prefix in their role name will be provisioned to SAP S/4HANA Cloud. Roles without this prefix in the role name won't be provisioned. If the property is not set, all roles will be provisioned to SAP S/4HANA Cloud.</li> </ul> <p><b>System Role:</b> Source and Target</p>	SAP S/4HANA Cloud

Name	Description	System Type
<code>ips.delete.threshold.users</code>	<p>Use this property to control the number of users to be deleted in a target system by defining a threshold. This will prevent you from accidentally deleting a huge number of users, for example by adding a filter or condition.</p> <p>The property is optional and is defined on the target systems only. It is not added at system creation. Its value must be greater than "0".</p> <p>Controlling the deletion works as follows:</p> <ol style="list-style-type: none"> <li>1. You add the <code>ips.delete.threshold.users</code> property on the target system before running a job and define a threshold value, for example: <code>500</code>.</li> <li>2. You run a read or resync job that is expected to delete a huge number of users.</li> </ol> <p>As a result, expect the following behavior:</p> <ul style="list-style-type: none"> <li>• If the number of the users to be deleted is lower or equal (for example: <code>400</code> or <code>500</code>) to the defined threshold value of <code>500</code>, Identity Provisioning continues with the deletion.</li> <li>• If the number of the users to be deleted is greater (for example: <code>1000</code>) than the defined threshold value of <code>500</code>, Identity Provisioning does not delete any users. Instead, the service marks the expected to be deleted users as failed in the job statistics. An error message in the job logs explains that users cannot be deleted as the defined threshold is exceeded.</li> </ul> <p><b>System Role:</b> Target</p>	All systems

Name	Description	System Type
<code>ips.delete.threshold.groups</code>	<p>Use this property to control the number of groups to be deleted in a target system by defining a threshold. This will prevent you from accidentally deleting a huge number of groups, for example by adding a filter or condition.</p> <p>The property is optional and is defined on the target systems only. It is not added at system creation. Its value must be greater than "0".</p> <p>Controlling the deletion works as follows:</p> <ol style="list-style-type: none"> <li>1. You add the <code>ips.delete.threshold.groups</code> property on the target system before running a job and define a threshold value, for example: <code>500</code>.</li> <li>2. You run a read or resync job that is expected to delete a huge number of groups.</li> </ol> <p>As a result, expect the following behavior:</p> <ul style="list-style-type: none"> <li>• If the number of the groups to be deleted is lower or equal (for example: <code>400</code> or <code>500</code>) to the defined threshold value of <code>500</code>, Identity Provisioning continues with the deletion.</li> <li>• If the number of the groups to be deleted is greater (for example: <code>1000</code>) than the defined threshold value of <code>500</code>, Identity Provisioning does not delete any groups. Instead, the service marks the expected to be deleted groups as failed in the job statistics. An error message in the job logs explains that groups cannot be deleted as the defined threshold is exceeded.</li> </ul>	<p>All, except for:</p> <ul style="list-style-type: none"> <li>• SAP BTP ABAP environment</li> <li>• SAP Application Server ABAP</li> <li>• SAP Integrated Business Planning for Supply Chain</li> <li>• SAP Market Communication</li> <li>• SAP Marketing Cloud</li> <li>• SAP S/4HANA Cloud</li> <li>• SAP S/4HANA On-Premise</li> </ul>
<b>System Role:</b> Target		

Name	Description	System Type
<code>ips.delete.threshold.roles</code>	<p>Use this property to control the number of roles to be deleted in a target system by defining a threshold. This will prevent you from accidentally deleting a huge number of roles, for example by adding a filter or condition.</p> <p>The property is optional and is defined on the target systems only. It is not added at system creation. Its value must be greater than "0".</p> <p>Controlling the deletion works as follows:</p> <ol style="list-style-type: none"> <li>1. You add the <code>ips.delete.threshold.roles</code> property on the target system before running a job and define a threshold value, for example: <code>500</code>.</li> <li>2. You run a read or resync job that is expected to delete a huge number of roles.</li> </ol> <p>As a result, expect the following behavior:</p> <ul style="list-style-type: none"> <li>• If the number of the roles to be deleted is lower or equal (for example: <code>400</code> or <code>500</code>) to the defined threshold value of <code>500</code>, Identity Provisioning continues with the deletion.</li> <li>• If the number of the roles to be deleted is greater (for example: <code>1000</code>) than the defined threshold value of <code>500</code>, Identity Provisioning does not delete any roles. Instead, the service marks the expected to be deleted roles as failed in the job statistics. An error message in the job logs explains that roles cannot be deleted as the defined threshold is exceeded.</li> </ul>	<p>SAP BTP ABAP environment</p> <p>SAP Application Server ABAP</p> <p>SAP Integrated Business Planning for Supply Chain</p> <p>SAP Market Communication</p> <p>SAP Marketing Cloud</p> <p>SAP S/4HANA Cloud</p> <p>SAP S/4HANA On-Premise</p>
<b>System Role:</b> Target		



Name	Description	System Type
<code>awf.domain</code>	<p>The domain name is the name of your SAP Advanced Workflow tenant.</p> <p>If you don't know your tenant name, contact your supervisor or administrator, or refer to the email notification you received when your account was created.</p> <p><b>System Role:</b> Source, Target, Proxy</p>	SAP Advanced Workflow
<code>awf.user.filter</code>	<p>When specified, only those SAP Advanced Workflow users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>userName eq "Julie Armstrong"</code></li> <li><code>userName sw "J"</code></li> <li><code>name.familyName eq "Armstrong"</code></li> <li><code>emails eq "julie.armstrong@example.com"</code></li> <li><code>userName eq "Julie Armstrong" and emails.value eq "julie.armstrong@example.com"</code></li> </ul> <p><b>System Role:</b> Source</p>	SAP Advanced Workflow
<code>awf.include.if.match.wildcard.header</code>	<p>Makes the connector send the <i>If-Match</i> HTTP header with a value of "*" for every request to the target system. This header could be used by an SAP Advanced Workflow system for entity versioning.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>true</code></li> <li><code>false</code></li> </ul> <p>Default value: <code>false</code></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Advanced Workflow

Name	Description	System Type
<code>awf.support.patch.operation</code>	<p>This property controls how modified users in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>If set to <i>true</i>, Identity Provisioning sends a PATCH request to the user or group resource in the target system. Only attributes without "scope": "createEntity" in the attribute mappings in the write transformation will be updated. For example, if the last name of a user is changed in the source system, the patch operation will update it in the target system and will leave unchanged other attributes with explicitly set "scope": "createEntity".</li> <li>If set to <i>false</i>, PUT operations are used to update users in the target system. This means, for example, that if a user attribute is modified, all user attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p>Users can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>In the source system, some user attributes are modified, or new attributes are added.</li> <li>In the source system, a condition or a filter is set for users not to be read anymore.</li> <li>A user is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <b>deleteEntity</b>.</p>	SAP Advanced Workflow

Name	Description	System Type
	<p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">true</a></li> <li><a href="#">false</a></li> </ul> <p>Default value: <a href="#">false</a></p> <p><b>System Role:</b> Target, Proxy</p>	
<code>awf.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists in SAP Advanced Workflow. Thus, the service needs to retrieve the <a href="#">entityId</a> of the existing user via filtering by user unique attribute(s). This property defines by which unique attribute(s) the existing user to be searched (resolved).</p> <p><b>Default behavior:</b> This property is missing during system creation. Its default value is <a href="#">userName</a>. That means, if the service finds an existing user by a <a href="#">userName</a>, it updates this user with the data of the conflicting one. If a user with such a <a href="#">userName</a> is not found, the creation of the conflicting user fails.</p> <p><b>Possible values:</b></p> <p>Default value: <a href="#">userName</a></p> <p><b>System Role:</b> Target, Proxy</p>	SAP Advanced Workflow

## 1.3.3 Transformations

Maintain the transformation logic, which corresponds to the structure and logic of your systems.

### What is a transformation?

For every system supported by the Identity Provisioning service, there is an initial (default) transformation logic that converts the system specific JSON representation of the entities from/to one common JSON. You can see it on the [Transformations](#) tab when you create a new system, after saving it. You can adjust the transformation mapping rules to reflect the current setup of entities from the source or target system.

## How it works

During the provisioning job, the source system reads its entities, and then using the configured read transformation, converts the source system specific JSON to the common JSON format. Then this common JSON format is passed to the target system, which applies the write transformation.

The administrator of the Identity Provisioning service can change this behavior by adapting the transformation logic to read only the entities that should be provisioned to the target system. This filter can speed up the processing of the entities and their provisioning to the target system.

## Transformation Editors

The Identity Provisioning service provides two editors for working with the transformation code: graphical editor and JSON (text) editor. The graphical editor is displayed by default. You can switch between them on the [Transformations](#) tab.

## How to use them

If you want to...	See
Configure the default system transformation	<a href="#">Transformation Editors [page 402]</a> <a href="#">Manage Transformations [page 1494]</a>
Learn more about read and write transformations	<a href="#">Transformation Types [page 324]</a>
See some examples	<a href="#">Transformation Examples [page 326]</a>
Learn about the supported functions	<a href="#">Transformation Functions [page 362]</a>
Learn which expressions you can use	<a href="#">Transformation Expressions [page 330]</a>
Learn how to use variables	<a href="#">Transformation Variables [page 399]</a>
Use parameters taken from a system's <a href="#">Properties</a> set	<a href="#">Properties [page 90]</a> → <b>Parameterized System Transformations</b>

### 1.3.3.1 Transformation Types

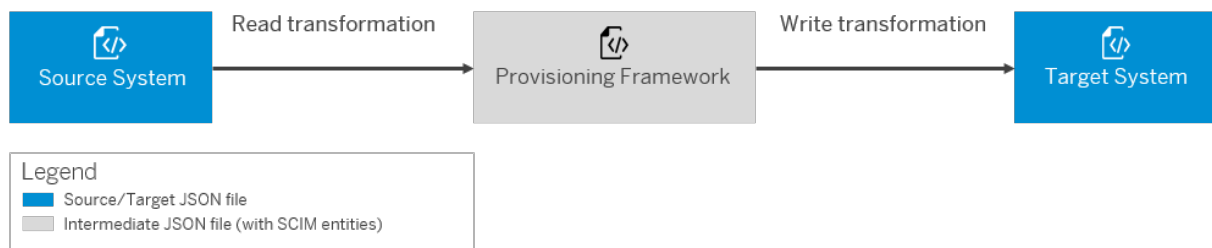
Learn about the types of JSON transformations needed for the provisioning jobs.

## Context

Two types of transformations occur before the provisioning of entities:

- **Read Transformation** – from the *source system* to the provisioning framework. It reads the data in the source system and transfers it to an intermediate JSON data in the provisioning framework. The reading of entities from the source system can be complete (full read) or partial (delta read). For more information, see [Manage Full and Delta Read \[page 1519\]](#).
- **Write Transformation** – from the provisioning framework to the *target system*. It prepares the data to be written to the target system.

Both transformations result in JSON data.



Every supported system holds and requires specific JSON data. To convert the source JSON data to an intermediate JSON version (which can be used for transformation to a supported target system), the Identity Provisioning administrator can use the suggested JSON transformation logic on the [Transformations](#) tab, and adapt it to the required transformation.

All transformations from the source systems transform their specific JSON data to intermediate data according to the [System for Cross-domain Identity Management](#) (SCIM 2.0) specification.

### Note

*Proxy systems* contain both *Read* and *Write* transformations.

## Example

If the source JSON data contains the attribute **name**, the read transformation converts this attribute to *name23* in the intermediate JSON data. Then, the write transformation should use the attribute *name23* (instead of **name**) as *sourcePath* attribute.

## Related Information

[Manage Transformations \[page 1494\]](#)

[Transformation Expressions \[page 330\]](#)

[Transformation Functions \[page 362\]](#)

## 1.3.3.2 Transformation Examples

The following examples explain how transformations work.

### Basic Transformation

The basic transformation copies all attributes from the input JSON messages to the output JSON ones.

#### Code Syntax

```
{
  "sourcePath": "$",
  "targetPath": "$"
},
```

### Source, Intermediate, and Target Data

In this example, the source system JSON data contains the **sn[0]** attribute. The read transformation converts this attribute to **name.familyName** in the intermediate JSON data. Then, the write transformation reads the **name.familyName** attribute and maps it to **name.familyName** in the target system.

#### Source entity data (from Microsoft Active Directory)

```
{
  "sAMAccountName": ["jsmith"],
  "mail": ["john.smith@company.com"],
  "givenName": ["John"],
  "sn": ["Smith"],
  "memberOf": ["group1"],
  "memberOf_2": ["group21", "group22"],
  "memberOf_3": ["group31", "group32", "group33"]
}
```

#### Read transformation (intermediate JSON Data)

```
{
  "user": {
    "mappings": [
      ...
      {
        "sourcePath": "$.ldap.attribute.user.id[0]",
        "targetVariable": "entityIdSourceSystem",
        "correlationAttribute": true
      },
      {
        "condition": "('%ldap.attribute.user.id%' == '%ldap.attribute.dn%')",
        "constant": "",
        "targetVariable": "nestedPathRegex",
        "functions": [
          {
            "function": "concatString",
            "prefix": "(?i)cn=.*?,(.*?)",

```

```

        "suffix": ",%ldap.user.path%"
      }
    ]
  },
  {
    "condition": "('%ldap.attribute.user.id%' ==
'%ldap.attribute.dn%')",
    "sourcePath": "$.%ldap.attribute.user.id%",
    "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:ad:2.0:User'] ['nestedPath']",
    "functions": [
      {
        "function": "getMatchedRegexGroup",
        "regex": "${nestedPathRegex}",
        "groupIndex": 1
      }
    ],
    "defaultValue": ""
  },
  {
    "sourcePath": "$.sAMAccountName[0]",
    "targetPath": "$.userName",
    "correlationAttribute": true
  },
  {
    "sourcePath": "$.displayName[0]",
    "optional": true,
    "targetPath": "$.displayName"
  },
  {
    "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
    "targetPath": "$.schemas[0]"
  },
  {
    "sourcePath": "$.mail[0]",
    "optional": true,
    "targetPath": "$.emails[0].value",
    "correlationAttribute": true
  },
  {
    "sourcePath": "$.givenName[0]",
    "optional": true,
    "targetPath": "$.name.givenName"
  },
  {
    "sourcePath": "$.sn[0]",
    "optional": true,
    "targetPath": "$.name.familyName"
  },
  {
    "sourcePath": "$.memberOf",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.groups[?(@.value)]"
  },
  ...

```

### Write transformation (intermediate JSON Data)

```

{
  "user": {
    "condition": "($.emails.length() > 0) && ($.name.familyName EMPTY
false)",
    "mappings": [
      {
        "sourcePath": "$.groups",
        "preserveArrayWithSingleElement": true,

```

```

        "optional": true,
        "targetPath": "$.corporateGroups"
    },
    {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
    },
    {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
    },
    {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": "$.schemas[1]"
    },
    {
        "sourcePath": "$.userName",
        "optional": true,
        "targetPath": "$.userName"
    },
    {
        "sourcePath": "$.emails[*].value",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails[?(@.value)]"
    },
    {
        "sourcePath": "$.userType",
        "optional": true,
        "targetPath": "$.userType"
    },
    {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
    },
    {
        "sourcePath": "$.name.middleName",
        "optional": true,
        "targetPath": "$.name.middleName"
    },
    {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.name.familyName"
    },
    {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
    },
    ...

```

#### Target entity data (result in Identity Authentication)

```

{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:User",
    "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
  ],
  "id": "P000100",
  "userName": "jsmith",
  "name": {
    "familyName": "Smith",
    "givenName": "John"
  },
  "emails": [

```



```

    {
      "value": "john.smith@company.com",
      "primary": "true",
      "type": "work"
    }
  ],
  "groups": [
    {
      "value": "group1"
    }
  ],
  "groups_2": [
    {
      "value": "group22"
    }
  ],
  "groups_3": [
    {
      "value": "group31"
    },
    {
      "value": "group32"
    },
    {
      "value": "group33"
    }
  ]
}

```

## Conditions in Transformations

In this example, you can try to apply a condition on whether an attribute of an entity contains a particular string. The template for such condition is the following:

```
"condition": "$.<attribute> =~ /.*<text>./",
```

where:

- `=~` means that a regular expression (regex) will be tested in the condition
- `/` represents the start and end of the regex
- `.*` represents any symbol
- `<text>` is a placeholder for the string that you want the value to contain

### ❖ Example

With this example, you can check if the first email address of a user contains a particular domain:

```
"condition": "$.emails[0].value =~ /.*@example.com/",
```

### ❖ Example

With this example, you can assign a user to a group, based on their *userName* containing a particular string:

```

{
  "condition": "$.userName =~ /.*explorer./",
  "constant": "Explorers",
  "targetPath": "$.groups[0].value"
}

```

```
    },  
    {  
      "condition": "$.userName =~ /.*scifi.*/",  
      "constant": "Scientists",  
      "targetPath": "$.groups[1].value"  
    }  
  ]  
}
```

**Result:** This will assign all users, which have "explorer" in their *userName*, to the "Explorers" group, and all users, which contain "scifi" in their *userName*, to the "Scientists" group.

## Related Information

[Manage Transformations \[page 1494\]](#)

### 1.3.3.3 Transformation Expressions

The transformation logic is based on JSON. The order of the expressions in the file is decisive for how the transformation is executed. The transformation actions are performed in the sequence defined in the transformation logic.

There is a different transformation logic for every entity (users, groups, roles). Below are listed some of the JSON expressions you can use.

## applyOnElements

Use this JSON expression when the value referenced by the [sourcePath](#) is a multivalue structure (a list or an array), and you want to apply a function to its elements instead of to the whole structure.

You can use the [applyOnElements](#) expression in any function, for any provisioning system.

For example, you can see it in the default write transformation of the [SAP Application Server ABAP](#) proxy system. In this case, the **decode** function reads all members of a user list and converts this list into a byte array. Then, the **toString** function reads this byte-array and converts it back into a string (a list of usernames).

## ❖ Example

```
// Function toString will be
// applied to all members read from a
// user list.
...
{
    "sourcePath":
    "$.members[*].value",

    "preserveArrayWithSingleElement":
    true,
    "optional": true,
    "targetPath": "$.USERLIST[?
    (@.USERNAME)]",
    "functions": [
        {
            "function":
            "decode",
            "algorithm":
            "base32",
            "skipPadding": true
        },
        {
            "function":
            "toString",
            "applyOnElements":
            true
        }
    ]
}
...
```

## applyOnJsonAttribute

This expression is deprecated. You can use the new one – [applyOnAttribute](#).

applyOnAttribute

Use this JSON expression when the structure referenced by the [sourcePath](#) is a complex one (a map), and you want to apply a function to a single member of this map.

In the example below, all occurrences of "SAP" in the values of attribute COMPOSITE\_PRIVILEGE are replaced with "test" in the transformation result.

### ❖ Example

Privileges in the source system:

```
{
  "PRIVILEGES": [
    {
      "COMPOSITE_PRIVILEGE":
      "SAP_01_COMP"
    },
    {
      "COMPOSITE_PRIVILEGE":
      "SAP_02_COMP"
    }
  ]
}
```

### ❖ Example

Source system transformation:

```
{
  "mappings": [
    {
      "sourcePath":
      "$.PRIVILEGES",
      "targetPath":
      "$.groups",
      "functions": [
        {
          "function":
          "replaceString",
          "applyOnElements": true,
          "target":
          "SAP",
          "replacement":
          "test",
          "applyOnAttribute":
          "COMPOSITE_PRIVILEGE"
        }
      ]
    }
  ]
}
```

### ❖ Example

Transformation result:

```
{
  "groups": [
    {
      "COMPOSITE_PRIVILEGE":
      "test_01_COMP"
    },
    {
      "COMPOSITE_PRIVILEGE":
      "test_02_COMP"
    }
  ]
}
```

assignToAttribute

Use this JSON expression when the structure referenced by [targetPath](#) is a complex one (a map), and you want to assign the result of this mapping to a specific map entry.

In the example below, all occurrences of "SAP" in the values of attribute COMPOSITE\_PRIVILEGE are replaced with "test", and the result is assigned to the value of attribute "ASSIGNMENT".

### ❖ Example

Privileges in the source system:

```
{
  "PRIVILEGES": [
    {
      "COMPOSITE_PRIVILEGE":
      "SAP_01_COMP"
    },
    {
      "COMPOSITE_PRIVILEGE":
      "SAP_02_COMP"
    }
  ]
}
```

### ❖ Example

Source system transformation:

```
{
  "mappings": [
    {
      "sourcePath":
      "$.PRIVILEGES",
      "targetPath":
      "$.groups",
      "functions": [
        {
          "function":
          "replaceString",
          "applyOnElements": true,
          "target":
          "SAP",
          "replacement":
          "test",
          "applyOnAttribute":
          "COMPOSITE_PRIVILEGE",
          "assignToAttribute": "ASSIGNMENT"
        }
      ]
    }
  ]
}
```

### ❖ Example

Transformation result:

```
{
  "groups": [
    {
      "ASSIGNMENT":
"test_01_COMP",
      "COMPOSITE_PRIVILEGE":
"SAP_01_COMP"
    },
    {
      "ASSIGNMENT":
"test_02_COMP",
      "COMPOSITE_PRIVILEGE":
"SAP_02_COMP"
    }
  ]
}
```

condition

A condition specifies a JSON filter expression. It can be applied on a single mapping entity or on the whole entity type. You can use conditions on strings, constants, variables and functions.

- Condition with **strings**

#### ❖ Example

```
{
  "mappings": [
    ...
    {
      "condition":
"$memberOf contains 'group1'",
      "constant":
"NewDisplayName",
      "targetPath": "$.displayName"
    }
  ]
}
```

- Condition with **constants** – AS ABAP (proxy system)

#### ❖ Example

```
{
  "condition":
"($.emails.length() > 0)
&& ($.name.familyName EMPTY
false)",
  "mappings": [
    {
      "sourcePath": "$",
      "targetPath": "$"
    },
    ...
  ]
}
```

- Condition with format **variables** – SAP S/4HANA Cloud (source system)

#### ❖ Example

```
{
  "user": {
    "condition":
"($.validityPeriod.startDate
<= '${currentDate}') &&
($.validityPeriod.endDate > '${currentDate}')",
    "mappings": [
      {

```



```

        "sourcePath":
        "$.personID",
        "targetVariable":
        "entityIdSourceSystem"
    },

```

- Conditions in **functions** – Microsoft Active Directory (target system)

#### ♣ Example

```

"group": {
  ...
  {
    "sourcePath":
    "$.members[*]",
    "targetVariable":
    "membersVariable",
    "preserveArrayWithSingleElement"
    : true,
    "optional": true,
    "functions": [
      {
        "condition":
        "@.type != 'Group'",
        "entityType":
        "user",
        "applyOnElements":
        true,
        "type":
        "resolveEntityIds"
      },
      {
        "condition":
        "@.type == 'Group'",
        "entityType":
        "group",
        "applyOnElements":
        true,
        "type":
        "resolveEntityIds"
      },
      {
        "condition":
        "(@.type != 'Group') &&
        ('%ldap.attribute.user.id%' !=
        '%ldap.attribute.dn%')",
        "function":
        "concatString",
        "applyOnElements":
        true,
        "applyOnAttribute":
        "value",
        "prefix":
        "%ldap.attribute.user.id%",
        "suffix":
        "%ldap.user.path%"
      }
    ]
  }
}

```

```

        },
        {
            "condition":
"@.type == 'Group') &&
('%ldap.attribute.group.id%' !=
'%ldap.attribute.dn%')",
            "function":
"concatString",
            "applyOnElements":
true,
            "applyOnAttribute":
"value",
            "prefix":
"%ldap.attribute.group.id%=",
            "suffix":
",%ldap.group.path%"
        }
    ]
    ...

```

- Conditions with **functions**

#### ❖ Example

```

{
  "user": {
    "condition": "($.emails
EMPTY false) &&
isValidEmail($.emails[0].value)"
  ,
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      .....
    ]
  }
}

```

constant

Set a constant if the target system requires attributes that are not defined in the source system.

You can also use schemas to organize and combine multiple constants.

#### ❖ Example

```

{
  "targetPath":
"$.emails[0].type",
  "constant": "work"
},

```

correlationAttribute

Correlation attributes are used in the *"user"* mappings of the [Read Transformations](#) (in source and proxy systems). They mark attributes that are unique for a user in multiple source systems where this user is read from.

The purpose of `correlationAttribute` is to properly count managed identities (users) during the reading process. This is essential because the calculated value is compared to the customer's licensing quota.

- [Use Case 1](#) – Same attribute in different systems

#### ❖ Example

Mapping from the [Microsoft Active Directory](#) read transformation:

```
{
  "sourcePath":
    "$.sAMAccountName[0]",
  "targetPath":
    "$.userName",
  "correlationAttribute":
    true
},
```

#### ❖ Example

Mapping from the [SAP S/4HANA On-Premise](#) read transformation:

```
{
  "sourcePath":
    "$.userAssignment.userID",
  "optional": true,
  "targetPath":
    "$.userName",
  "correlationAttribute":
    true
},
```

#### ❖ Example

Mapping from the [LDAP Server](#) read transformation:

```
{
  "sourcePath":
    "$.ldap.attribute.user.id[0]",
  "targetPath":
    "$.userName",
  "correlationAttribute":
    true
},
```

```
},
```

**Result:** If the Identity Provisioning reads a user whose username is one and the same in all the three systems (as listed above), the service will consider and count these "3 users" as one managed identity.

- [Use Case 2](#) – Different attributes from several systems

#### ❖ Example

Mapping from the [Microsoft Active Directory](#) read transformation:

```
{
  "sourcePath":
    "$.sAMAccountName[0]",
  "targetPath":
    "$.userName",
  "correlationAttribute":
    true
},
```

#### ❖ Example

Mapping from the [SAP Analytics Cloud](#) read transformation:

```
{
  "sourcePath": "$.emails",
  "targetPath": "$.emails",

  "preserveArrayWithSingleElement"
  : true
},
{
  "sourcePath": "$.emails[?
(@.primary== true)].value",
  "correlationAttribute":
    true
},
```

#### ❖ Example

Mapping from the [Identity Authentication](#) read transformation:

```
{
  "sourcePath":
    "$.userName",
  "targetPath":
    "$.userName",
  "optional": true,
  "correlationAttribute":
    true
},
```

```
    },  
    ...  
    {  
      "sourcePath":  
        "$.emails[*].value",  
      "preserveArrayWithSingleElement"  
        : true,  
      "targetPath": "$.emails[?  
        (@.value)]"  
    },  
    {  
      "sourcePath": "$.emails[?  
        (@.primary== true)].value",  
      "correlationAttribute":  
        true  
    },  
  ],  
}
```

**Result:** If the Identity Provisioning reads a user with a certain username from one system, then it reads a user with a particular e-mail from another system, and then finds a user with the same username and e-mail in a third system (as listed above), the service will consider and count these "3 users" as one managed identity.

**Possible values:**

- *true*
- *false*

**Data type:** Boolean

defaultValue

It returns the default value of an attribute when no specific functions or conditions are set for this attribute. The **defaultValue** expression comes in handy in the following cases:

- When an attribute of a previously provisioned entity is later deleted from the source system. After a new provisioning job, the target system will try to get some value for this missing attribute, and thus it will write its default one.
- When the system transformation contains a [valueMapping](#) operation. If the value of the mapped attribute is not found as a key in the "valueMappings" definitions, or some of the source paths does not return a value, the JSON transformation will use the default one.

The default value of an attribute can be a [string](#), [integer](#), [Boolean](#), [array](#), or empty (null).

You can add "defaultValue" for any attribute, in the transformation of any provisioning system. For example, you can find it in the default transformation of the **Microsoft Active Directory** target system:

#### ❖ Example

```
...
{
  "sourcePath":
"$$.displayName",
  "optional": true,
  "targetPath":
"$$.displayName[0]",
  "defaultValue": []
},
{
  "sourcePath":
"$$.emails[0].value",
  "optional": true,
  "targetPath": "$$.mail[0]",
  "defaultValue": []
},
{
  "sourcePath":
"$$.name.givenName",
  "optional": true,
  "targetPath":
"$$.givenName[0]",
  "defaultValue": []
},
...
}
```

ignore

Use the *ignore* expression if you prefer parts of the transformation to not be taken into consideration (during provisioning). Similar to *condition*, you can set *ignore* on various levels – for a whole entity type (user, role) or for a particular mapping entity. This is applicable for both source and target systems.

#### ❖ Example

```
"group": {
  "ignore": true,
  "mappings": [
    {
      "sourcePath":
        "$.sAMAccountName[0]",
      "targetVariable":
        "entityIdSourceSystem"
    },
    {
      "sourcePath":
        "$.sAMAccountName[0]",
      "targetPath":
        "$.displayName"
    },
    ...
  ]
}
```

#### ❖ Example

```
"user": {
  "mappings": [
    {
      "ignore": true,
      "sourcePath":
        "$.sAMAccountName[0]",
      "targetVariable":
        "entityIdSourceSystem"
    },
    {
      "sourcePath":
        "$.sAMAccountName[0]",
      "targetPath":
        "$.userName"
    },
    ...
  ]
}
```

#### Possible values:

- *true*
- *false*

**Data type:** Boolean

## preserveArrayWithSingleElement

This JSON expression converts an array into a single element. That means, when the Identity Provisioning service reads an array that contains only one element, then in the target system:

- If **"preserveArrayWithSingleElement"** is set to *true*, this expression will keep the array with the single element as is.

## ❖ Example

The following mapping keeps the array of group members, resolving their IDs, even if the array contains only one member.

```
{
  "sourcePath":
    "$.members[*].value",
  "preserveArrayWithSingleElement"
    : true,
    "optional": true,
    "targetPath": "$.members[?
(@.value)]",
    "functions": [
      {
        "type":
        "resolveEntityIds"
      }
    ]
}
```

- If **"preserveArrayWithSingleElement"** is set to *false*, this expression will convert the array into the relevant single element.

## ❖ Example

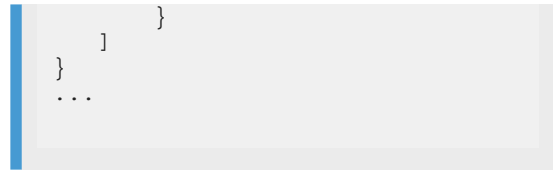
The following mapping converts the array into a single element (a group member), and resolves its ID.

```
{
  "sourcePath":
    "$.members[*].value",
  "preserveArrayWithSingleElement"
    : false,
    "optional": true,
    "targetPath": "$.members[?
(@.value)]",
    "functions": [
      {
        "type":
        "resolveEntityIds"
      }
    ]
}
```



## Transformation Expression

## Usage



### Possible values:

- *true*
- *false*

**Data type:** Boolean

scope

You can set a scope for an entity attribute, based on its lifecycle. A scope can have the following values:

- **createEntity** – an entity attribute is only processed during creation. To do this, tag the entity attribute with scope `createEntity` in the system transformation. Transformation mappings without scope are always processed.

### i Note

Currently, the `createEntity` scope is only applicable for entities created in **target systems**.

### ♣ Example

The following mapping provides an initial password when a user is created.

```
{
  "user": {
    "mappings": [
      {
        "scope":
"createEntity",
        "targetPath":
"$ .Password",
        "constant":
"Initial1"
      }
    ]
  },
  ...
}
```

- **patchEntity** – when using the *ipsproxy* application and a system that supports SCIM PATCH operation, you may need to perform certain transformations over the PATCH request. To do this, use the `patchEntity` scope in the system transformation. Only mappings with this scope will be processed.

### i Note

- The Identity Provisioning service supports PATCH requests only for groups and group members (assignments).
- The `patchEntity` scope is only applicable for write transformations in **proxy systems**.

### ❖ Example

The following mapping decodes the group ID and preserves the patch request body (payload) into the target system.

```
{
  "scope": "patchEntity",
  "sourceVariable":
"entityIdTargetSystem",
  "targetVariable":
"entityIdTargetSystem",
  "functions": [
    {
      "type": "decode",
      "algorithm":
"base32",
      "skipPadding": true
    },
    {
      "function":
"toString"
    }
  ]
},
{
  "scope": "patchEntity",
  "sourcePath": "$",
  "targetPath": "$"
},
}
```

- **deleteEntity** – If an entity has been deleted from the source system or has been set a condition for it not to be read anymore, this entity can "stay" in the target system for the following reasons:
  - The target system does not support deletion of entities.
  - You do not want to delete it but only temporary disable/deactivate it.
  - You want to neither delete it, nor deactivate it but only remove its permissions, or exclude it from some corporate groups.

If you have to fulfill some of these scenarios for an entity, use the `deleteEntity` scope. It prevents from deleting the entity from the target system as only updating its status instead. Also, bear in mind the following:

- For the affected entity, all transformation mappings that do not contain this scope will be ignored.
- If a condition exists on entity type level, it will be ignored as well.

- Use this scope for *SCIM* systems, as well as *Concur*, *Microsoft Azure AD*, *Identity Authentication*, and *SAP Jam*.

### ❖ Example

**Concur:** The following mapping disables the user account:

```
{
  "user": {
    "mappings": [
      {
        "scope":
"deleteEntity",
        "constant": "US",
        "targetPath":
"$$.Custom21"
      },
      {
        "scope":
"deleteEntity",
        "constant": "",
        "targetPath":
"$$.Password"
      },
      {
        "scope":
"deleteEntity",
        "constant": "DEFAULT",
        "targetPath":
"$$.LedgerCode"
      },
      {
        "constant": "N",
        "targetPath":
"$$.Active",
        "scope": "deleteEntity"
      },
      ...
    ]
  }
}
```

### ❖ Example

**Microsoft Azure AD:** The following mapping disables the user account:

```
{
  "user": {
    "mappings": [
      {
        "constant": false
        "targetPath":
"$$.accountEnabled",
        "scope":
"deleteEntity",
      },
    ]
  }
}
```

...

### ❖ Example

**Identity Authentication (SCIM API version 1):** The following mapping disables the user account and unassigns the user from all the groups it was a member of:

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id",
        "scope": "deleteEntity"
      },
      {
        "constant": false,
        "targetPath":
"$$.active",
        "scope": "deleteEntity"
      },
      {
        "constant": [],
        "targetPath":
"$$.corporateGroups",
        "scope": "deleteEntity"
      },
      {
        "constant": [],
        "targetPath":
"$$.groups",
        "scope": "deleteEntity"
      }
    ]
  }
  ...
}
```

### ❖ Example

**Identity Authentication (SCIM API version 2):** The following mapping disables the user account.

Disabling a user account is done with a PATCH operation which replaces the value of the active user attribute **true** with **false**. For this, you also need to set

`ias.user.update.instead.delete=true` in the Identity Authentication target system.

For more information, see: [List of Properties \[page 94\]](#)

{

```

    "user": {
      "mappings": [
        {
          "constant": "urn:ietf:params:scim:api:messages:2.0:PatchOp",
          "targetPath": "$.schemas[0]",
          "scope": "deleteEntity"
        },
        {
          "constant": "replace",
          "targetPath": "$.Operations[0].op",
          "scope": "deleteEntity"
        },
        {
          "constant": "active",
          "targetPath": "$.Operations[0].path",
          "scope": "deleteEntity"
        },
        {
          "constant": false,
          "targetPath": "$.Operations[0].value",
          "scope": "deleteEntity"
        },
        ...
      ]
    }

```

### ❖ Example

**SAP Jam:** The following mapping disables the user account:

```

    "user": {
      "mappings": [
        {
          "sourceVariable": "entityIdTargetSystem",
          "targetPath": "$.id",
          "scope": "deleteEntity"
        },
        {
          "constant": false,
          "targetPath": "$.active",
          "scope": "deleteEntity"
        }
      ]
    }

```

## Transformation Expression

## Usage

```
"deleteEntity"      "scope":  
    },  
    ...
```

### Possible values:

- [createEntity](#)
- [patchEntity](#)
- [deleteEntity](#)

**Data type:** String

## skipOperations

If you want the provisioning job to not execute operations of a certain type on groups and users, use the `skipOperations` expression. You can apply it when you need to avoid creating, deleting, or updating entities.

You can use `skipOperations` only in target system transformations.

- In the following example, the transformation does not allow updating groups in the target system. This approach is valuable when you have done group changes in the target system, and after a new [Read](#) or [Resync](#) job, you want these changes to be kept instead of being overwritten by the groups in the source system.

## ❖ Example

```
"group": {
  "skipOperations": [
    "update"
  ],
  "mappings": [
    {
```

- In the following example, the transformation does not allow creating and deleting users in the target system:

## ❖ Example

```
{
  "user": {
    "skipOperations": [
      "create", "delete"
    ],
    "mappings": [
      {
```

You can set the transformation to skip the "create" operation if you're sure that all entities from the source system already exist in the target – independently from the Identity Provisioning service. That means, they have been created in the target system manually or via a script. Thus, by skipping the "create" operation, the identity provisioning job will only provision the changes instead of creating the users all over again in the target. The Identity Provisioning service will skip creating the existing entities and will try to update them. There are two options for resolving the existing users by the service.

**Option 1:** If your target system supports PATCH operation and it is enabled, the Identity Provisioning service



will search and resolve the existing users by the default or predefined unique attributes for the system. For more information about the unique attributes of your system, see [List of Properties \[page 94\]](#).

**Option 2:** The existing users are resolved by retrieving their IDs. In order for the *skip create* operation to be executed without error, you also need to make a few adjustments in your source and target transformations. This way, the service will get and retrieve the IDs of the entities correctly.

### **i** Note

If an entity does not exist in the target system (which means, a retrieved ID does not match any target entity), this entity will neither be created, nor updated.

- In the following detailed example, *SAP SuccessFactors* is a source system and *Identity Authentication* is a target system. You want all users that exist on both systems to be skipped during creation and to only be updated on the target. If there are users existing only in the source system, they will not be created in the target. To do that:

1. Open your *SAP SuccessFactors* admin console. For every user that you want to be provisioned and updated in *Identity Authentication*, you need to set up one of its predefined custom attributes (*custom01* – *custom15*), entering the *Identity Authentication* unique identifier for that user. For example, you can do that in the **custom10** attribute.

### **i** Note

The unique identifier that you have to set up for the relevant SAP SuccessFactors users depends on the API version of your Identity Authentication system.

- For Identity Authentication SCIM API (in short, *SCIM API version 1*), enter the **User ID** (P-user).
- For Identity Directory SCIM API (in short, *SCIM API version 2*), enter the **User UUID**.

2. Then go to the *Identity Provisioning* admin console, and in the *Properties* tab update the `sf.user.attributes` property, adding **custom10** to the list of attributes.

3. In the *SAP SuccessFactors* source system, add the following extra user mapping. It maps **custom10** to a custom attribute in *Identity Authentication*, for example **custom\_IAS\_ID**:

#### ❖ Example

```
{
  "user": {
    "mappings": [
      {
        "sourcePath":
        "$.custom10",
        "targetPath":
        "$.custom_IAS_ID",
        "optional": true
      },
      ...
    ]
  }
}
```

4. In the *Identity Authentication* target system, replace the following user mapping:

#### ❖ Example

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable":
        "entityIdTargetSystem",
        "targetPath":
        "$.id"
      },
      ...
    ]
  }
}
```

with:

#### ❖ Example

```
{
  "sourcePath":
  "$.custom_IAS_ID",
  "targetVariable":
  "entityIdTargetSystem"
},
{
  "sourcePath":
  "$.custom_IAS_ID",
  "targetPath": "$.id"
}
```

```
},
```

- (Optional) If there are users that exist only in the source system and you don't want them to be created in the target, enhance the following condition in the *Identity Authentication* target transformation:

#### ❖ Example

```
{
  "user": {
    "condition":
      "($.emails.length() > 0)
      && ($.name.familyName EMPTY
      false) && ($.custom_IAS_ID
      EMPTY false)",
    "mappings": [
      . . .
    ]
  }
}
```

- Run the first *Read* provisioning job.
- Check in the *Identity Authentication* admin console if the relevant "marked" users are updated, and that the rest are not. Also, check that users that exist only in *SAP SuccessFactors* (if any) are not created in *Identity Authentication*.

#### Possible values:

- create*
- update*
- delete*

**Data type:** String

sourcePath

The *sourcePath* expression denotes the path to an attribute in the input JSON message (could be the source system JSON data or the intermediate one).

#### ❖ Example

```
{
  "sourcePath":
    "$.name.familyName",
  "targetPath": "$.name.familyName"
},
```

## Transformation Expression

## Usage

targetPath

The *targetPath* expression denotes the path where the attribute should be stored in the output JSON message (could be the intermediate or the target system JSON data).

### ❖ Example

```
{
  "targetPath":
    "$.name.familyName",
  "sourcePath": "$.sn[0]"
},
```

type

The type of action to be performed in the mapping. It could be [set](#), [remove](#), [rename](#) and [valueMapping](#).

- The [set](#) type maps an attribute from the source system to an attribute in the target JSON data. If no type is defined, "type": "set" is used by default.

#### ❖ Example

```
{
  "type": "set",
  "targetPath": "$.groups"
}
```

- The [remove](#) type deletes an attribute during transformation. This attribute is not present in the target JSON data.

#### ❖ Example

```
{
  "type": "remove",
  "targetPath": "$.groups"
}
```

- The [rename](#) type renames the last element of a complex targetPath.

In the example below, the last element `.lastName` of the targetPath `$.name.lastName` will be renamed with the provided constant value `familyName`. This will result in the following targetPath:  
`$.name.familyName",.`

#### ≡ Code Syntax

```
{
  "targetPath":
  "$.name.lastName",
  "constant":
  "familyName",
  "type":
  "rename",
  "optional": true
}
```

- The [valueMapping](#) type allows multiple entity attributes (read from the source system) to be mapped to a single target attribute.  
 For example, you can set a mapping condition for a user attribute **user.timeZoneCode**. After the provisioning job, its value will be mapped to a new attribute –

**timezone.** The example below provides a number of world locations and their relevant timezone.

### Note

If the value of **user.timeZoneCode** is not found as a key in the **valueMappings** definitions, or some of the source paths does not return a value, the JSON transformation will use the default one. (In the example below, the default value is *Europe/Berlin*.)

### Example

JSON code for mapping user timezone:

```
"user":
{
  ...

  {
    "type":
    "valueMapping",
    "sourcePaths":
    [ "$.user.timeZoneCode" ],
    "targetPath":
    "$.timezone",
    "defaultValue":
    "Europe/Berlin",
    "valueMappings": [
      { "key":
      [ "WDFI" ], "mappedValue":
      "Europe/Berlin" },
      { "key":
      [ "ISRAEL" ], "mappedValue":
      "Asia/Jerusalem" },
      { "key":
      [ "RUS03" ], "mappedValue":
      "Europe/Moscow" },
      { "key":
      [ "AUSNSW" ], "mappedValue":
      "Australia/Sydney" },
      { "key":
      [ "UTC+4" ], "mappedValue": "Asia/
      Dubai" },
      { "key":
      [ "BRAZIL" ], "mappedValue":
      "America/Sao_Paulo" },
      { "key":
      [ "BRZLEA" ], "mappedValue":
      "America/Sao_Paulo" },
      { "key":
      [ "MSTNO" ], "mappedValue":
      "America/Phoenix" },
      { "key":
      [ "EST" ], "mappedValue":
      "America/New_York" },
```

```
        { "key":  
          [ "UTC" ], "mappedValue": "Etc/  
          UTC" },  
        { "key":  
          [ "UTC+3" ], "mappedValue": "Asia/  
          Riyadh" },  
        { "key":  
          [ "EST_" ], "mappedValue":  
          "America/Toronto" },  
        { "key":  
          [ "UTC+8" ], "mappedValue": "Asia/  
          Shanghai" },  
        { "key":  
          [ "JAPAN" ], "mappedValue": "Asia/  
          Tokyo" }  
      ]  
    }  
    ...  
  }
```

**Possible values:**

- [set](#)
- [remove](#)
- [rename](#)
- [valueMapping](#)

**Data type:** String

## valueMappings

Provides a list with mapped values. It can only be used when *type* is set to **valueMapping**. The list contains the keys, which are read from the source system and mapped to values in a single target attribute.

The example below provides the list of world locations and their relevant timezone.

## ❖ Example

JSON code for mapping user timezone:

```
"user":
{
  ...
  {
    "type": "valueMapping",
    "sourcePaths":
    [ "$.user.timeZoneCode" ],
    "targetPath":
    "$.timezone",
    "defaultValue":
    "Europe/Berlin",
    "valueMappings": [
      { "key":
      [ "WDFT" ], "mappedValue": "Europe/
      Berlin" },
      { "key":
      [ "ISRAEL" ], "mappedValue": "Asia/
      Jerusalem" },
      { "key":
      [ "RUS03" ], "mappedValue": "Europe/
      Moscow" },
      { "key":
      [ "AUSNSW" ], "mappedValue":
      "Australia/Sydney" },
      { "key":
      [ "UTC+4" ], "mappedValue": "Asia/
      Dubai" },
      { "key":
      [ "BRAZIL" ], "mappedValue":
      "America/Sao_Paulo" },
      { "key":
      [ "BRZLEA" ], "mappedValue":
      "America/Sao_Paulo" },
      { "key":
      [ "MSTNO" ], "mappedValue": "America/
      Phoenix" },
      { "key":
      [ "EST" ], "mappedValue": "America/
      New_York" },
      { "key":
      [ "UTC" ], "mappedValue": "Etc/
      UTC" },
      { "key":
      [ "UTC+3" ], "mappedValue": "Asia/
      Riyadh" },
```



```

    { "key":
      [ "EST_" ], "mappedValue": "America/
      Toronto" },
    { "key":
      [ "UTC+8" ], "mappedValue": "Asia/
      Shanghai" },
    { "key":
      [ "JAPAN" ], "mappedValue": "Asia/
      Tokyo" }
    ]
  }
  ...

```

sourcePaths

Specifies the paths to an attribute in the input JSON message of the source system. The *sourcePaths* can only be used when *type* is set to **valueMapping**.

optional

Specifies whether attribute's availability is mandatory or optional for a given system.

- If set to *true*, this means that the attribute is optional and even if it is missing, provisioning won't fail.
- If set to *false* or not set at all, this means that the attribute is mandatory and if it is missing, provisioning will fail.

For example, the default read transformation of SAP SuccessFactors shows that if the first name of a user is missing in the source system, this user will be read and provisioned to a target system. However, if the last name of a user is missing, provisioning will fail as this is a mandatory attribute.

#### Code Syntax

```

{
  "sourcePath":
  "$.firstName",
  "optional": true,
  "targetPath":
  "$.name.givenName"
},
{
  "sourcePath": "$.lastName",
  "targetPath":
  "$.name.familyName"
},

```

Possible values:

- *true*
- *false*

Data type: Boolean

Transformation Expression	Usage
sourceVariable	<p>The <i>sourceVariable</i> expression denotes a variable that is valid only within a given provisioning system. It shows where the variable will be read from.</p> <p>For more information, see <a href="#">Transformation Variables [page 399]</a></p>
targetVariable	<p>The <i>targetVariable</i> expression denotes a variable that is valid only within a given provisioning system. It shows where the variable will be written to.</p> <p>For more information, see <a href="#">Transformation Variables [page 399]</a></p>

### 1.3.3.4 Transformation Functions

A function is a hardcoded piece of transformation logic that receives a value denoted by the input specified by a source path or a source variable. As a result, the value is replicated into the target path or target variable, accordingly.

The Identity Provisioning service uses two types of functions: *conditional* [\[page 363\]](#) and *mapping* [\[page 365\]](#) functions. The conditional functions are functions which can be used only in conditional statements. The mapping functions are used in entity transformations and are included as mappings.

Functions can also be chained – that means, the output of one function is the input for the next one. See an example with such functions in the **encode / decode** section below.

For more information, see the tables below.

## Conditional Functions

Function	Parameters	Description
isValidEmail	<ul style="list-style-type: none"> <li>JSON path attribute which holds the email value Required: Yes Type: String</li> <li>Possible values: <ul style="list-style-type: none"> <li><code>\$.emails[0].value</code></li> <li><code>emails.value</code></li> </ul> </li> </ul>	<p>This function verifies whether an e-mail address is valid by checking if the given String value matches the following regex pattern:</p> <pre>"[a-zA-Z0-9!#\$%&amp;\'*/=?^_`{ }~-]+(?:\\.[a-zA-Z0-9!#\$%&amp;\'*/=?^_`{ }~-]+)*@(?:[a-zA-Z0-9!#\$%&amp;\'*/=?^_`{ }~-]+\\.)+[a-zA-Z0-9!#\$%&amp;\'*/=?^_`{ }~-]+(?:[a-zA-Z0-9!#\$%&amp;\'*/=?^_`{ }~-]+)"</pre> <p>After the check, the function returns a Boolean result.</p> <p>This function can be used only as part of a conditional statement.</p> <p>EXAMPLE:</p> <p>In this example, the source JSON code contains the following e-mail address of a user:</p> <pre>{   "userName": "test",   "emails": [     {       "value": "example@company.com",       "primary": true     }   ] }</pre> <p>The condition with function <a href="#">isValidEmail</a> is the following:</p> <pre>{   "condition": "isValidEmail(\$.emails[0].value)",   "mappings": [</pre>

Function	Parameters	Description
		<pre> {   "sourcePath":   "\$.userName",   "targetPath":   "\$.userName" }, {   "sourcePath":   "\$.emails[*].value",   "preserveArrayWithSingleElement": true,   "targetPath":   "\$.emails[?   (@.value)]" } ] } </pre>

## Mapping Functions

Function	Parameters	Description
concatString	<ul style="list-style-type: none"> <li> <b>prefix</b>  Required: No  Type: String </li> <li> <b>suffix</b>  Required: No  Type: Integer; String; Boolean </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function concatenates a string with a prefix or a suffix.</p> <p>EXAMPLE 1:</p> <p>In the following transformation example, function <i>concatString</i> will be applied to all source logon names (SAM-account user names) read from <i>Microsoft AD</i>, adhering a prefix <b>_ips</b> and a suffix <b>123</b> in the target system. For example, a <i>sAMAccountName</i> name <b>johnsmith</b> from the source system will be provisioned as <b>ips_johnsmith123</b> in the target system.</p> <div> <p>❖ Example</p> <pre> {   "user": {     "mappings": [       {         "sourcePath":         "\$.sAMAccountName[0]",         "targetPath":         "\$.userName",         "functions": [           {             "function":             "concatString",             "prefix": "_ips_",             "suffix": 123           }         ]       }     ]   } } </pre> </div> <p>EXAMPLE 2:</p>

Function	Parameters	Description
		<p>In the following transformation example, function <code>concatString</code> will be applied to all source user IDs in a source system, converting them into uniform e-mails in the <i>Microsoft Azure AD</i> target system. In this case, the <code>concatString</code> function takes the value of the <code>aad.domain.name</code> property (which is the name of a verified Microsoft Azure AD domain) and adheres it to the <code>userId</code>. For example, a source user ID <code>johnsmith123</code> can produce a target principal name <code>johnsmith123@mail.acme.com</code>.</p> <div> <div>❖ Example</div> <pre> {   "sourcePath":     "\$.userId",   "targetPath":     "\$.emails[0].value",   "correlationAttribute": true,   "functions": [     {       "function":         "concatString",       "suffix":         "@%aad.domain.name%"     }   ],   ... }</pre> </div>



Function	Parameters	Description
convertCountryRegion	<ul style="list-style-type: none"> <li> <b>outputFormat</b>  Required: Yes  Possible values: <ul style="list-style-type: none"> <li>fullName</li> <li>alpha2</li> </ul> </li> <li> <b>inputAttributes</b>  Required: Yes  Possible values: <ul style="list-style-type: none"> <li>country</li> <li>region</li> </ul> </li> <li> <b>outputAttribute</b>  Required: Yes  Possible values: region </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function converts the countries into format compliant with <a href="#">ISO 3166-2</a>.</p> <pre> {   "condition":     "(\$.addresses[*].region EMPTY false) &amp;&amp; (\$.addresses[*].country EMPTY false)",   "sourcePath":     "\$.addresses",   "preserveArrayWithSingleElement": true,   "optional":     true,   "targetPath":     "\$.addresses",   "functions": [     {       "type":         "convertCountryRegion",       "outputFormat":         "alpha2",       "inputAttributes":         [ "region", "country" ]     }   ] } </pre> <p>EXAMPLE:</p> <p>In the following transformation example, function <a href="#">convertCountryRegion</a> will be applied to all source regions read from the source system, converting them in the wanted ISO format. For example, the region name <a href="#">US-AL</a> or only <a href="#">AL</a> from the source system will be provisioned to the target system as:</p> <ul style="list-style-type: none"> <li><a href="#">US-AL</a> if the outputFormat is <a href="#">alpha2</a>;</li> </ul>



Function	Parameters	Description
		<ul style="list-style-type: none"> <li><i>Alabama</i> if the <code>outputFormat</code> is <i>fullName</i>.</li> </ul>

Function	Parameters	Description
compositeld	<ul style="list-style-type: none"> <li> <b>separator</b>  Required: Yes  Possible values: : (colon) </li> <li> <b>subId</b>  Required: Yes  Type: String; JSONPath expression </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function appends the value referred by the source path to the value of the subId parameter, separated by ":".</p> <div> <p>❖ Example</p> <p>Input JSON code before transformation:</p> <pre>{   "qualifierId":     "johnsmith" }</pre> </div> <div> <p>❖ Example</p> <p>Transformation using the function:</p> <pre>{   "sourcePath":     "\$.qualifierId",   "targetPath":     "\$     ['composite_id.cu     stom_separator.co     lon']",   "functions": [     {       "function":         "compositeId",       "separator": ":",       "subId":         "e0123456"     }   ],   ... }</pre> </div> <div> <p>❖ Example</p> <p>After the function execution, the output JSON results to:</p> <pre>{   "composite_id.def   ault_separator":     "johnsmith:e01234     56",   ... }</pre> </div>

Function	Parameters	Description
		<pre> "composite_id.custom_separator.column": "johnsmith:e0123456",  "composite_id.custom_separator.dash": "johnsmith- e0123456" } </pre>

Function	Parameters	Description
copyMapEntry	<ul style="list-style-type: none"> <li> <b>sourceKey</b>  Required: Yes  Type: String </li> <li> <b>targetKey</b>  Required: Yes  Type: String </li> </ul>	<p>This function copies a key-value pair from the source JSON and creates a similar one in the target JSON. The newly created pair consists of the specified <code>targetKey</code> String and the value copied from the initial pair. The two key-value pairs are written into the target system.</p> <p>EXAMPLE:</p> <p>For example, if the source entity contains the following roles:</p> <pre>{ "role": [ { "roleName": "ROLE1" }, { "roleName": "ROLE2" } ] }</pre> <p>After applying the function:</p> <pre>{ "sourcePath": "\$ .user.role", "optional": true, "targetPath": "\$ .groups", "preserveArrayWithSingleElement": true, "functions": [ { "function": "copyMapEntry", "sourceKey": "roleName", "targetKey": "value", "value" } ] }</pre> <p>Two pairs with <code>targetKey</code> <b>"value"</b> are added to the target JSON and the result is:</p> <pre>{ "groups": [ { "roleName": "ROLE1", "value": "ROLE1" }, { "roleName": "ROLE2", "value": "ROLE2" } ] }</pre>

Function	Parameters	Description
decode	<ul style="list-style-type: none"> <li> <b>algorithm</b>  Required: Yes  Possible values: <ul style="list-style-type: none"> <li>base32</li> <li>base64</li> </ul> </li> <li> <b>skipPadding</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> Default value: false </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>The function <b>decode</b> is part of the Write transformation of some proxy systems (like AS ABAP). It decodes the string value specified by the sourcePath or sourceVariable mapping attribute, by using specified decoding algorithm.</p> <p><b>Function result:</b> Byte array</p> <p>EXAMPLE:</p> <p>For example, if the input JSON code contains the following user name:</p> <pre>{   "USERNAME": "JOHN*SMITH/Junior" }</pre> <p>The base32 encoded result will be the following id:</p> <p><b>❖ Example</b></p> <pre>{   "id": "JJHUQTRKKGUSVCIF5FHK3TJN5ZA====" }</pre> <p>The <i>decode</i> function reads the <b>id</b> and converts it into a byte array. Afterward, the <i>toString</i> function reads this byte array and converts it into the original <b>USERNAME</b> string. The base32 padding symbols (==) are skipped.</p> <p><b>❖ Example</b></p> <pre>// Write Transformation {   "user": {     "mappings": [       {         "sourcePath": "\$.id", </pre>

Function	Parameters	Description
		<pre> "targetPath": "\$ .USERNAME",  "functions": [    {      "function":     "decode",      "algorithm":     "base32",      "skipPadding":     true    },    {      "function":     "toString"    }  ],  ... </pre>

Function	Parameters	Description
encode	<ul style="list-style-type: none"> <li> <b>algorithm</b>  Required: Yes  Possible values: <ul style="list-style-type: none"> <li>base32</li> <li>base64</li> </ul> </li> <li> <b>skipPadding</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> Default value: false </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>The function <b>encode</b> is part of the Read transformation of some proxy systems (like AS ABAP). It encodes the value specified by the sourcePath or sourceVariable mapping attribute, by using specified encoding algorithm.</p> <p>The function can accept a Byte array or everything that has a string representation. The string is assumed in UTF-8 character encoding.</p> <p><b>Function result:</b> String</p> <p>EXAMPLE:</p> <p>For example, if the input JSON code contains the following user name:</p> <pre>{   "USERNAME":   "JOHN*SMITH/Junior" }</pre> <p>The base32 encoded result will be the following id:</p> <p><b>❖ Example</b></p> <pre>{   "id":   "JJHUQTRKKGUSVCIF5F   HK3TJN5ZA=== " }</pre> <p>The <a href="#">encode</a> function reads the <b>USERNAME</b> string and writes the encoded value into the <b>id</b> path. The base32 padding symbols (==) are skipped.</p> <p><b>❖ Example</b></p> <pre>// Read Transformation {   "user": {</pre>

Function	Parameters	Description
		<pre> "mappings": [   {     "sourcePath":     "\$.USERNAME",     "targetPath":     "\$.id",     "targetVariable":     "entityIdSourceSy stem",     "functions": [       {         "function":         "encode",         "algorithm":         "base32",         "skipPadding":         true       }     ],     ... </pre>



Function	Parameters	Description
getMatchedRegexGroup	<ul style="list-style-type: none"> <li> <b>regex</b>  Required: Yes  Type: String </li> <li> <b>groupIndex</b>  Required: Yes  Type: Integer </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function checks whether a given string matches the provided regular expression (regex), and returns the required group.</p> <p>For example, when the value of groupIndex is:</p> <ul style="list-style-type: none"> <li>0 – the function returns the entire string</li> <li>Positive number – the function returns a specific group from the string. For instance, groupIndex = 3 will return the third group.</li> </ul> <p>To learn more about the relevant Java method, see: <a href="#">Java API: Class Matcher – group (int)</a> ➡</p> <div> <p>❖ Example</p> <p>Initial user data:</p> <pre>{   "distinguishedName": [     "CN=test10,OU=subUsers,OU=testUsers"   ] }</pre> </div> <div> <p>❖ Example</p> <p>Transformation in the source system:</p> <pre>{   "group": {     "mappings": [       {         ...       }     ]   },   "sourcePath":     "\$.distinguishedName[0]",   "functions": [     { </pre> </div>

Function	Parameters	Description
		<pre> "function": "getMatchedRegexGroup",  "regex": "(? i)cn=.*?, (.*)OU=testUsers",  "groupIndex": 1     }     ],  "targetPath": "\$ ['urn:sap:cloud:s cim:schemas:exten sion:ad:2.0:User' ]['nestedPath']"     }     ]     } </pre>
		<p>❖ <b>Example</b></p> <p>Nested path extracted from the user DN, after the transformation:</p> <pre> {  "urn:sap:cloud:sc im:schemas:extens ion:ad:2.0:User": {  "nestedPath": "OU=subUsers"     } } </pre>

Function	Parameters	Description
manipulateDate	<ul style="list-style-type: none"> <li> <b>sourceDateFormat</b>  Required: No  Type: Date; Time </li> <li> <b>targetDateFormat</b>  Required: No  Type: Date; Time </li> <li> <b>years</b>  Required: No  Type: Integer </li> <li> <b>months</b>  Required: No  Type: Integer </li> <li> <b>days</b>  Required: No  Type: Integer </li> <li> <b>hours</b>  Required: No  Type: Integer </li> <li> <b>minutes</b>  Required: No  Type: Integer </li> <li> <b>seconds</b>  Required: No  Type: Integer </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function converts one date format into another after JSON-Path transformations.</p> <p>Use cases:</p> <ul style="list-style-type: none"> <li>A Java date format can be converted into another Java date format. Example: "<b>2018-02-28 11:00:00.000</b>" to "<b>02/28/2018</b>"</li> <li>A date format based on <a href="#">Unix Time Stamp</a> can be converted into a Java one. That means, if the source system stores a date as a number of milliseconds, after the transformations this number will be converted and written in the target system as a human readable date. Example: "<b>Date(1519809649123-0240)</b>" to "<b>2018-02-28 UTC+1</b>"</li> </ul> <div> <p><b>i Note</b></p> <p>Bear in mind the following restrictions about <a href="#">Unix Time Stamp</a> format:</p> <ul style="list-style-type: none"> <li>It is mainly applicable for SAP SuccessFactors connectors.</li> <li>If the source date format contains a timezone (GMT, EST, ACT, etc.), after converting from <a href="#">Unix Time Stamp</a>, the date will be displayed as a UTC offset.</li> <li>During calculation, the timezone is ignored – the milliseconds are converted to a "pure"</li> </ul> </div>

Function	Parameters	Description
		<p>date. The timezone is displayed (as UTC offset) but not taken into account.</p> <p>The <i>manipulateDate</i> function supports the following operations:</p> <ul style="list-style-type: none"> <li>• <b>(Java)</b> Incrementing the date by the "+" sign or when there is no sign</li> <li>• <b>(Java)</b> Decrementing the date by the "-" sign</li> <li>• <b>(Unix Time Stamp)</b> Converting a number of milliseconds into a human readable date</li> </ul> <p><b>i Note</b></p> <p>The parameters <code>sourceDateFormat</code> and <code>targetDateFormat</code> accept values in short or full date &amp; time format. For example: <code>yyyy-MM-dd'T'HH:mm:ss'Z'</code>.</p> <p><b>❖ Example</b></p> <p><b>Reads and writes the current date in standard Java date format</b></p> <pre>{   "targetPath":     "\$.EmployeeType.ValidityPeriod.StartDate",   "sourceVariable":     "currentDate",   "functions":     [       {         "function":           "manipulateDate",         "targetDateFormat"           : "MM/dd/yyyy",         "sourceDateFormat"           : "yyyy-MM-dd           HH:mm:ss.SSS",</pre>

Function	Parameters	Description
		<pre> "years":  "months": // You can also, for example, increment the date with 3 days and 2 hours  "days": "3"  "hours": "+2"  "minutes":  "seconds": } ... } </pre>
		<p>❖ <b>Example</b></p> <p><b>Reads a given date in Unix Time Stamp format (in milliseconds) and writes the converted value in the target system as a standard Java date format</b></p> <pre> { "targetPath": "\$ .EmployeeType.V alidityPeriod.Sta rtDate", "sourcePath": "\$date", "functions": [ { "function": "manipulateDate", "sourceDateFormat": "Date(milliseconds)", "targetDateFormat": "yyyy-MM-dd HH:mm:ss.SSS" } ... } </pre>

Function	Parameters	Description
		<pre>} </pre>
matchRegex	<ul style="list-style-type: none"> <li> <b>regex</b>  Required: Yes  Type: String </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function checks whether a given string matches the provided regular expression and returns a Boolean result.</p> <p>❖ <b>Example</b></p> <p>Checks whether the value of attribute "COMPOSITE_PRIVILEGE" matches the regular expression "*_COMP", and assigns the result (true or false) to attribute "composite".</p> <pre>{   "mappings": [     {       "sourcePath":         "\$.PRIVILEGES",       "targetPath":         "\$.groups",       "functions": [         {           "function":             "matchRegex",           "regex":             ".*_COMP",           "applyOnAttribute":             "COMPOSITE_PRIVILEGE",           "assignToAttribute":             "composite"         }       ]     }   ] }</pre>

Function	Parameters	Description
putIfAbsent	<ul style="list-style-type: none"> <li> <b>key</b>  Required: Yes  Type: String </li> <li> <b>defaultValue</b>  Required: Yes  Type: String; Array; Integer; another value type </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function works on user attributes of type array (list of elements), as each element consists of key-value pairs. If an element key misses a value, the putIfAbsent function will set a default value for this key.</p> <div> <div>❖ Example</div> <p>This function is part of the default <b>Identity Authentication</b> target and proxy write transformation. In this case, the user attribute "<b>addresses</b>" is a list of addresses. The <b>putIfAbsent</b> function is applied on every single address (element) of this list.</p> <pre> ... {   "sourcePath":     "\$.addresses",   "targetPath":     "\$.addresses",   "preserveArrayWithSingleElement":     true,   "defaultValue":     [],   "optional": true,   "functions": [     {       "function":         "putIfAbsent",       "key": "type",       "defaultValue":         "work"     }   ],   ... } </pre> </div>

Function	Parameters	Description
putIfPresent	<ul style="list-style-type: none"> <li>• <b>key</b> Required: Yes Type: String</li> <li>• <b>defaultValue</b> Required: Yes Type: String; Array; Integer; another value type</li> <li>• <b>condition</b> Required: No Type: String</li> <li>• <b>applyOnElements</b> Required: No Type: Boolean Possible values: <ul style="list-style-type: none"> <li>• true</li> <li>• false</li> </ul> </li> <li>• <b>applyOnAttribute</b> Required: No Type: String</li> <li>• <b>assignToAttribute</b> Required: No Type: String</li> </ul>	<p>This function works on user attributes of type <b>array</b> (list of elements), as each element consists of <i>key-value</i> pairs.</p> <ul style="list-style-type: none"> <li>• If an element key has a value but it's different than the default one, the <b>putIfPresent</b> function will set a default value for this key.</li> <li>• If an element has a value and it's equal to the default one, the <b>putIfPresent</b> function keeps this value as is.</li> </ul> <div> <div>❖ Example</div> <p>This function is part of the default <b>Identity Authentication</b> target and proxy write transformation. In this case, the user attribute "<b>addresses</b>" is a list of addresses. The <b>putIfPresent</b> function is applied on every single address (element) of this list.</p> <pre> ... {   "sourcePath":     "\$.addresses",   "targetPath":     "\$.addresses",   "preserveArrayWithSingleElement":     true,   "defaultValue":     [],   "optional": true,   "functions": [     {       "condition":         "(@.type         NIN ['work',         'home'])",       "function":         "putIfPresent", </pre> </div>



Function	Parameters	Description
		<pre> "key": "type", "defaultValue": "work"     }   ], }, ... </pre>

Function	Parameters	Description
randomPassword	<ul style="list-style-type: none"> <li> <b>passwordLength</b>  Required: Yes  Type: Integer </li> <li> <b>minimumNumberOfLowercaseLetters</b>  Required: Yes  Type: Integer </li> <li> <b>minimumNumberOfUppercaseLetters</b>  Required: Yes  Type: Integer </li> <li> <b>minimumNumberOfDigits</b>  Required: Yes  Type: Integer </li> <li> <b>minimumNumberOfSpecialSymbols</b>  Required: Yes  Type: Integer </li> <li> <b>specialSymbols</b>  Required: No  Possible values: List of allowed special symbols (no comma separation)  Default value: ~!@#\$%^&amp;*()_+ </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function generates a random password. It picks characters from four character sets - digits, lowercase letters, uppercase letters, and special symbols.</p> <p>Bear in mind the following tips:</p> <ul style="list-style-type: none"> <li>The password length must be supplied along with the number of characters from each set. If a value "0" is supplied for a given parameter, no characters will be picked from the corresponding character set.</li> <li>If the summed up number of characters (from all sets) exceeds the total password length, the function execution will result in error.</li> <li>If the summed up number of characters (from all sets) is less than the total password length, the remaining characters will be randomly picked from all character sets.</li> <li>A custom character set is supplied by the <code>specialSymbols</code> parameter.</li> <li>If a custom set of special symbols is supplied, the parameter <code>minimumNumberOfSpecialSymbols</code> cannot have a value of "0".</li> </ul> <div> <p><b>Note</b></p> <p>The <code>randomPassword</code> function does not require <code>sourcePath</code>, <code>sourceVariable</code>, or constant to be specified in the mapping.</p> </div> <div> <p><b>Example</b></p> </div>

Function	Parameters	Description
		<pre> {   "user": {     "mappings": [       {         "targetPath":         "\$.password",         "functions": [           {             "function":             "randomPassword",             "passwordLength":             16,             "minimumNumberOfL owercaseLetters":             4,             "minimumNumberOfU ppercaseLetters":             4,             "minimumNumberOfD igits": 4,             "minimumNumberOfS pecialSymbols":             4,             "specialSymbols":             ",.&lt;.&gt;/?~`!@#"           }         ]       }     ]   } } </pre>

Function	Parameters	Description
renameMapEntry	<ul style="list-style-type: none"> <li> <b>sourceKey</b>  Required: Yes  Type: String </li> <li> <b>targetKey</b>  Required: Yes  Type: String </li> </ul>	<p>This function reads the <code>sourceKey</code> String from the source JSON and renames it with the given <code>targetKey</code> String in the target JSON.</p> <p>EXAMPLE:</p> <p>For example, if the source entity contains the following roles:</p> <pre>{   "role": [     { "roleName": "ROLE1" },     { "roleName": "ROLE2" } ] }</pre> <p>After applying the function:</p> <pre>{   "sourcePath": "\$.user.role",   "optional": true,   "targetPath": "\$.groups",   "preserveArrayWithSingleElement": true,   "functions": [     {       "function": "renameMapEntry",       "sourceKey": "roleName",       "targetKey": "value",       "value": ""     }   ] }</pre> <p>The <code>sourceKey</code> String in the target JSON is replaced with the one given in the <code>targetKey</code> and the result is:</p> <pre>{   "groups": [     { "value": "ROLE1" },     { "value": "ROLE2" } ] }</pre>

Function	Parameters	Description
replaceString	<ul style="list-style-type: none"> <li> <b>target</b>  Required: Yes  Type: String; number </li> <li> <b>replacement</b>  Required: Yes  Type: String; number </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function reads strings from the source system and then replaces each substring that matches the value of the target parameter with the replacement one.</p> <p>For example, let's say the source system contains a string <i>"National College of Linguistics and History"</i>. The target substring is <b>"National"</b> and the replacement substring is <b>"European"</b>. Then, in the target system the main string will be written as <i>"European College of Linguistics and History"</i>.</p> <div> <div>❖ Example</div> <pre> {   "user": {     "mappings": [       {         "sourcePath":         "\$.sAMAccountName[0]",         "targetPath":         "\$.userName",         "functions": [           {             "function":             "replaceString",             "target": "iag",             "replacement":             "ips"           }         ]       }     ]   } } </pre> </div>

Function	Parameters	Description
replaceFirstString	<ul style="list-style-type: none"> <li> <b>regex</b>  Required: Yes  Type: String; number </li> <li> <b>replacement</b>  Required: Yes  Type: String; number </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function replaces the first substring of a given string that matches the provided regex with the string in replacement.</p> <div> <div>❖ Example</div> <pre> {   "user": {     "mappings": [       {         "sourcePath":         "\$.sAMAccountName[0]",         "targetPath":         "\$.userName",         "functions": [           {             "function":             "replaceFirstString",             "regex": "14\\d{1}",             "replacement":             123           }         ]       }     ]   } } </pre> </div>

Function	Parameters	Description
replaceLastString	<ul style="list-style-type: none"> <li> <b>regex</b>  Required: Yes  Type: String; number </li> <li> <b>replacement</b>  Required: Yes  Type: String; number </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function replaces the last substring of a given string that matches the provided regex with the string in replacement.</p> <div> ❖ <b>Example</b> <pre> {   "user": {     "mappings": [       {         "sourcePath": "\$sAMAccountName[0]",         "targetPath": "\$user.userName",         "functions": [           {             "function": "replaceLastString",             "regex": "14\\d{1}",             "replacement": 123           }         ]       }     ]   } } </pre> </div>

Function	Parameters	Description
replaceAllString	<ul style="list-style-type: none"> <li> <b>regex</b>  Required: Yes  Type: String; number </li> <li> <b>replacement</b>  Required: Yes  Type: String; number </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function replaces each substring of the given string that matches the provided regex with the string in replacement.</p> <div> ❖ <b>Example</b> <pre> {   "user": {     "mappings": [       {         "sourcePath": "\$sAMAccountName[0]",         "targetPath": "\$user.userName",         "functions": [           {             "function": "replaceAllString",             "regex": "14\\d{1}",             "replacement": 123           }         ]       }     ]   } } </pre> </div>



Function	Parameters	Description
resolveEntityIds	<p><b>entityType</b></p> <p>Required: No</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• user</li> <li>• group</li> </ul> <p>Default value: user</p> <ul style="list-style-type: none"> <li>• <b>condition</b> Required: No Type: String</li> <li>• <b>applyOnElements</b> Required: No Type: Boolean Possible values: <ul style="list-style-type: none"> <li>• true</li> <li>• false</li> </ul> </li> <li>• <b>applyOnAttribute</b> Required: No Type: String</li> <li>• <b>assignToAttribute</b> Required: No Type: String</li> <li>• <b>sourceSystemName</b> Required: No Type: String</li> <li>• <b>targetSystemName</b> Required: No Type: String</li> </ul>	<p>This function resolves the value of a source system attribute to an existing back-end key in the target system.</p> <p>For example, it can resolve the value of a source system member attribute to the ID of an existing SCIM resource that represents this member in a SCIM target system.</p> <div> <p>❖ Example</p> <pre> {   "sourcePath":     "\$.member",   "preserveArrayWithSingleElement":     true,     "optional":     true,   "targetPath":     "\$.members[?       (@.value)],",     "functions":     [       {         "entityType":           "group",         "function":           "resolveEntityIds"       }     ]     ... } </pre> </div> <p>The <code>sourceSystemName</code> and the <code>targetSystemName</code> parameters are typically used within this function to resolve group members.</p> <p>In the following example, we assume that you have created two source systems reading entities from one and the same user</p>

Function	Parameters	Description
		<p>store (for example, MS Azure AD). You provision users from <a href="#">SourceSystem1</a> and groups from <a href="#">SourceSystem2</a> to one and the same target system. By specifying the sourceSystemName as <a href="#">SourceSystem1</a>, the user IDs of the group members in <a href="#">SourceSystem2</a> will be resolved against the user IDs in the <a href="#">SourceSystem1</a>.</p> <div> <pre> ≡ Code Syntax  "functions": [   {     "sourceSystemName":     "SourceSystem1",     "function":     "resolveEntityIds"   } ] ... </pre> </div>

Function	Parameters	Description
splitStringToArray	<ul style="list-style-type: none"> <li><b>separator</b> Required: Yes Type: String</li> </ul>	<p>This function splits the initial String using the defined separator and produces as a result an array of Strings or objects.</p> <p>In the following examples, function <i>splitStringToArray</i> is applied to the <code>sourcePath</code>, read from the source system. Based on the <code>targetPath</code> defined, the initial String will be split into array of Strings or objects.</p> <p><b>Example 1</b></p> <p>In the first example, the initial String we use is:</p> <pre>"countries": "Serbia,Germany,Spain"</pre> <p>The transformation using the function is the following:</p> <pre>{   "sourcePath":   "\$.countries",   "targetPath":   "\$.countryList",   "functions": [     {       "function":       "splitStringToArray",       "separator":       ", "     }   ] }</pre> <p>After the function execution, the output results to the following array of Strings:</p> <pre>"countryList": [   "Serbia",   "Germany",</pre>

Function	Parameters	Description
		<pre>], "Spain"</pre>
		<p><b>Example 2</b></p> <p>In the second example, we use the same initial String:</p> <pre>"countries": "Serbia,Germany,Spain"</pre> <p>In this case, the <code>targetPath</code> in the transformation using the function supports multiple values:</p> <pre>{   "sourcePath":   "\$.countries",   "targetPath":   "\$.countryList[?   (@.name)]",   "functions": [     {       "function":       "splitStringToArray",       "separator":       ", "     }   ] }</pre> <p>After the function execution, the output results to the following array of objects:</p> <pre>"countryList": [   {     "name":     "Serbia"   },   {     "name":     "Germany"   } ]</pre>

Function	Parameters	Description
		<pre> "Spain"      "name":   } ], </pre>
substring	<ul style="list-style-type: none"> <li> <b>beginIndex</b>  Required: Yes  Type: Integer </li> <li> <b>endIndex</b>  Required: No  Type: Integer </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function returns a string if endIndex is not provided. It begins at the specified beginIndex and extends either to the character at index endIndex - 1 or to the end of this string.</p> <div> ❖ Example <pre> {   "user": {     "mappings": [       {         "sourcePath": "\$\$.sAMAccountName[0]",         "targetPath": "\$\$.userName",         "functions": [           {             "function": "substring",             "beginIndex": 3,             "endIndex": "5"           }         ]       }     ]   } } </pre> </div>

Function	Parameters	Description
toString	<ul style="list-style-type: none"> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function accepts a Byte array or a string representation. The string is assumed in UTF-8 character encoding. Function result: <b>String</b>.</p> <p>To see a relevant example, go to function encode and decode.</p>
toUpperCaseString	<ul style="list-style-type: none"> <li> <b>locale</b>  Required: No  Possible values: Abbreviation of locale, such as de_DE, ja_JP, ru_RU and so on.  Default value: en_EN </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function converts all the characters in the given string to upper case, using the provided locale, or if nothing defined – English.</p> <div> <div>❖ Example</div> <pre> {   "user": {     "mappings": [       {         "sourcePath":           "\$.SAMAccountName[0]",         "targetPath":           "\$.userName",         "functions": [           {             "function":               "toUpperCaseString",             "locale": "en_EN"           }         ]       }     ]   }   ... }</pre> </div>

Function	Parameters	Description
toLowerCaseString	<ul style="list-style-type: none"> <li> <b>locale</b>  Required: No  Possible values: Abbreviation of locale, such as de_DE, ja_JP, ru_RU and so on.  Default value: en_EN </li> <li> <b>condition</b>  Required: No  Type: String </li> <li> <b>applyOnElements</b>  Required: No  Type: Boolean  Possible values: <ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul> </li> <li> <b>applyOnAttribute</b>  Required: No  Type: String </li> <li> <b>assignToAttribute</b>  Required: No  Type: String </li> </ul>	<p>This function converts all the characters in the given string to lower case, using the provided locale, or if nothing defined – English.</p> <div> <div>♣ Example</div> <pre> {   "user": {     "mappings": [       {         "sourcePath":           "\$.sAMAccountName[0]",         "targetPath":           "\$.userName",         "functions": [           {             "function":               "toLowerCaseString",             "locale": "en_EN"           }         ]       }     ]   }   ... }</pre> </div>

### 1.3.3.5 Transformation Variables

System variables specify particular attributes of the read and written entities. They help you map attributes between source and target transformations so that the entities are provisioned correctly to the target systems.

Variable	Definition & Example	Mandatory in <systems>
entityIdSourceSystem	<p>Mandatory for every read transformation (in source and proxy systems). It specifies which attribute of a read entity to be considered as a unique ID in the source system.</p> <p>Use this mapping with caution, as it can overwrite the value of the ID returned by the source system.</p> <div> ❖ Example <pre> {   "targetVariable": "entityIdSourceSystem",   "sourcePath": "\$.name" } </pre> </div>	<ul style="list-style-type: none"> <li>Source systems</li> <li>Proxy systems</li> </ul>
entityIdTargetSystem	<p>Mandatory for every write transformation (in target and proxy systems). It specifies which attribute of a written entity to be considered as a unique ID in the target system. This variable is defined by the target system according to the system response during entity creation, or is read from the Identity Provisioning database during entity modification or deletion.</p> <div> ❖ Example <pre> {   "scope": "deleteEntity",   "sourceVariable": "entityIdTargetSystem",   "targetVariable": "entityIdTargetSystem",   "functions": [     {       "type": "decode",       "algorithm": "base32",       "skipPadding": true     },     {       "type": "toString"     }   ] } </pre> </div>	<ul style="list-style-type: none"> <li>Target systems</li> <li>Proxy systems</li> </ul>



Variable	Definition & Example	Mandatory in <systems>
entityBaseLocation	<p>Mandatory only for read transformations in proxy systems. It contains the proxy application URL featuring the entity type endpoint:</p> <p><b>https://ipsproxy&lt;proxy_provider_account&gt;-&lt;consumer_account&gt;.&lt;neo_landscape&gt;:443/ipsproxy/api/v1/scim/&lt;system_ID&gt;/Users</b></p> <div> <p>❖ Example</p> <pre> {   "sourceVariable": "entityBaseLocation",   "targetVariable": "entityLocationSourceSystem",   "targetPath": "\$.meta.location",   "functions": [     {       "type": "concatString",       "suffix": "\${entityIdSourceSystem}"     }   ] }</pre> </div>	Proxy systems
entityLocationSource-System	<p>Mandatory only for read transformations in proxy systems. It contains the proxy application URL featuring the SCIM 2.0 resource endpoint for an entity:</p> <p><b>https://ipsproxy&lt;proxy_provider_account&gt;-&lt;consumer_account&gt;.&lt;neo_landscape&gt;:443/ipsproxy/api/v1/scim/&lt;system_ID&gt;/Users/&lt;user_ID&gt;</b></p> <div> <p>❖ Example</p> <pre> {   "sourceVariable": "entityBaseLocation",   "targetVariable": "entityLocationSourceSystem",   "targetPath": "\$.meta.location",   "functions": [     {       "type": "concatString",       "suffix": "\${entityIdSourceSystem}"     }   ] }</pre> </div>	Proxy systems

Variable	Definition & Example	Mandatory in <systems>
currentDate	<p>An optional variable, which contains the current date, in format: <code>yyyy-MM-dd HH:mm:ss.SSS</code></p> <p>You can configure this variable by using property <code>ips.date.variable.format</code>, according to the <a href="#">Java Class DateTimeFormatter</a> .</p> <div> <p>❖ Example</p> <pre> {   "targetPath":     "\$.PersonalDetails.ValidityPeriod.StartDate"   ,   "sourceVariable": "currentDate",   "functions": [     {       "type": "manipulateDate",       "targetDateFormat": "yyyy-MM-dd"     }   ] }</pre> </div>	All systems

### 1.3.3.6 Transformation Editors

Identity Provisioning provides graphical and JSON text editor for managing provisioning system transformations.

#### Graphical Editor

This editor allows you to graphically model the entities (users, groups, roles) and their content (attribute mappings and transformation expressions, like conditions, functions, constants, and others). It offers the following advantages:

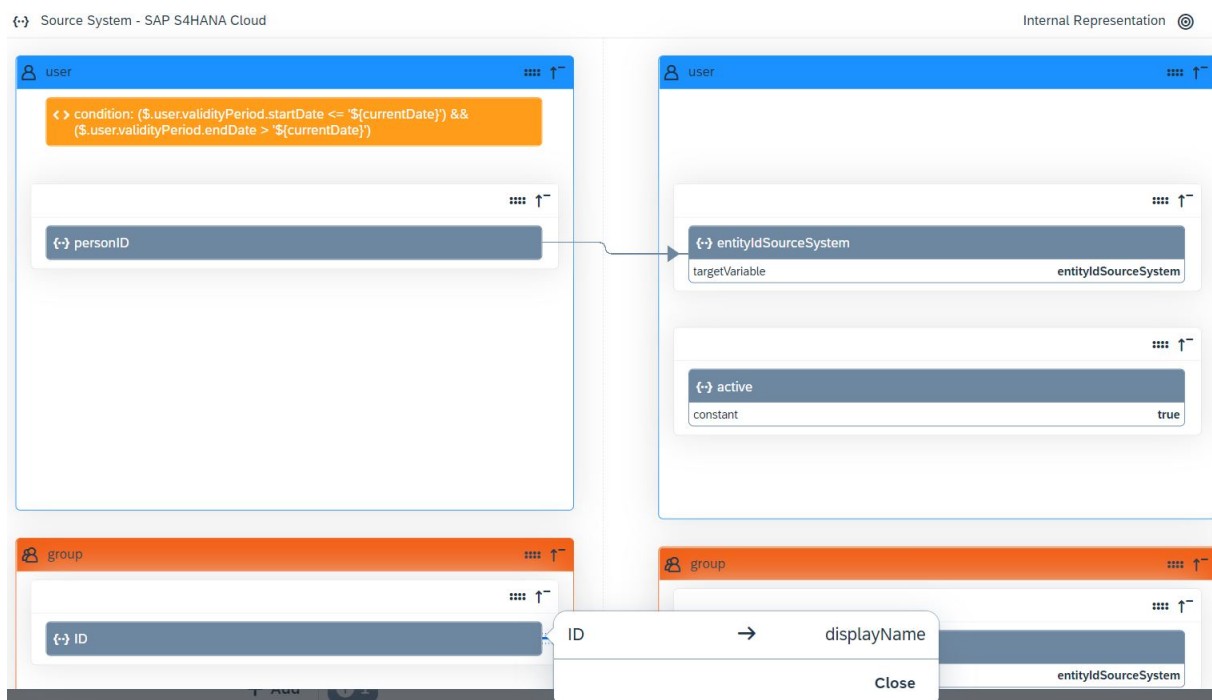
- **Visualization**  
Entities are organized in colored boxes. Every entity has a dedicated color. Users are always blue, groups are orange and roles are green. Source and target path attributes are displayed horizontally (side-by-side). They are connected with an arrow which indicates the direction of reading and writing the data. Color coding makes it easy to identify specific configurations. Ignored entities are greyed out, skipped operations and conditions are displayed in orange.
- **Simplification**  
Typing code and following JSON syntax rules are no longer needed, except in cases when conditions are defined. Expressions and functions along with their possible values are prefilled and available for selection in dropdown lists. The editor takes care of adding the dot notation, that is, separating child elements with a period (.), enclosing string values within double quotes, prepending the dollar sign (\$) to the JSONPath expression.

- Validation  
The JSON validator and the preview ensure that the input is in the expected format.

## i Note

The graphical editor is available only for Identity Provisioning tenants running on SAP Cloud Identity infrastructure. It is the default editor.

The following interactive screenshot shows the main elements of the graphical editor. Hover over the image and click highlighted areas for more information.

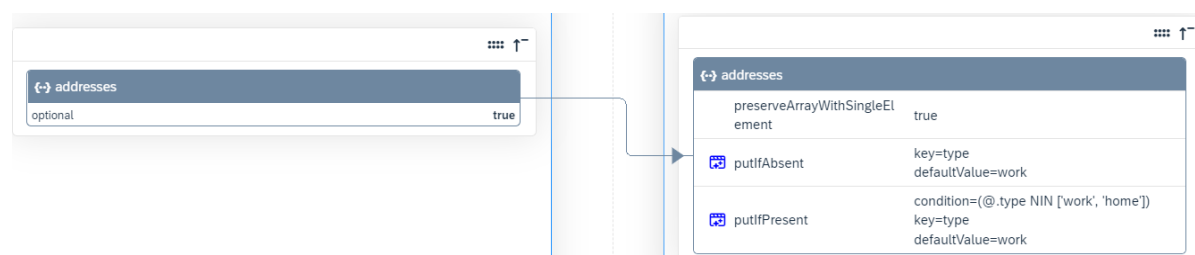


- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im0 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im8 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im1 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im9 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im2 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im4 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im3 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im10 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im11 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im5 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im12 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im13 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im6 \[page 404\]](#)
- [#unique\\_86/unique\\_86\\_Connect\\_42\\_subsection-im7 \[page 404\]](#)

Hover over each element for a description. Click the element for more information.

The following example illustrates how the address attribute mapping from the Identity Authentication write transformation is displayed in the graphical editor. Compare the same attribute with its representation in the JSON editor below.

## Attribute Mapping in Graphical Editor



The source path attribute **addresses** in the internal representation (on the left-hand side) and the target path attribute **addresses** in the target system (on the right-hand side) are displayed horizontally. The arrow shows the direction of reading and writing the data. The transformation expressions and functions are listed under the respective `sourcePath` and `targetPath` attribute they are applicable for.

## JSON Editor

This editor allows you to work with entities and their transformation mappings in text mode. To add or modify transformations, you need to type. You can also perform operations like select, cut, copy, paste and others. The editor provides line numbering, syntax coloring and syntax validation.

The following example illustrates how the address attribute mapping from the Identity Authentication write transformation is displayed in the JSON editor:

### Attribute Mapping in JSON Editor

The source path attribute **addresses** in the internal representation and the target path attribute **addresses** in the target system are displayed vertically - one below the other.

The transformation expressions and functions are listed in individual lines below the respective `sourcePath` and `targetPath` attribute they are applicable for.

Syntax coloring and syntax validation are enabled by default.

```
64 {
65   "sourcePath": "$.addresses",
66   "preserveArrayWithSingleElement": true,
67   "optional": true,
68   "targetPath": "$.addresses",
69   "functions": [
70     {
71       "function": "putIfAbsent",
72       "key": "type",
73       "defaultValue": "work"
74     },
75     {
76       "function": "putIfPresent",
77       "condition": "(@.type NIN ['work', 'home'])",
78       "key": "type",
79       "defaultValue": "work"
80     }
81   ]
82 }
```

Although the graphical and the JSON editors differ in the user experience they provide, other tasks, such as: export and reset transformations remain the same.

## Related Information

[Manage Transformations \[page 1494\]](#)

[Working with Graphical Editor \[page 1495\]](#)

## 1.4 Initial Setup of Bundle Tenants

Set up the systems for your provisioning scenario and run the provisioning jobs.

### Prerequisites

- You have purchased an SAP cloud solution bundled with Identity Authentication and Identity Provisioning services. For more information, see [Obtain a Bundle Tenant \[page 407\]](#).
- Bundle tenants created after March 15, 2022, run on the infrastructure of SAP Cloud Identity Services. Bundle tenants created before March 15, 2022, run on SAP BTP, Neo environment. For more information, see [Tenant Infrastructure \[page 10\]](#)

### Procedure

1. Log on to the Identity Provisioning user interface (UI). For more information, see [Access your Identity Provisioning bundle tenant \[page 413\]](#).
2. Set up the source, target, and proxy systems for your provisioning scenario. For more information on the availability of provisioning systems in bundle tenants, see [Provisioning Systems for Bundle Tenants \[page 416\]](#)

In bundle tenants, some source and target systems are preconfigured. This means, the connection to the relevant source and target systems is automatically set up and you can run provisioning jobs. To add and configure more systems relevant for your bundle, see [Add a System \[page 1477\]](#).

When adding a system, Identity Provisioning works as follows:

- Identity Provisioning in **default mode**: Provision user data from source to target systems. You add source and target systems only. This could be one source connected to one or multiple target systems, or one target connected to one or multiple source systems. For more information, see: [Source Systems \[page 452\]](#) and [Target Systems \[page 702\]](#).
  - Identity Provisioning in **proxy mode**: Provision user data to and from a central identity management solution and a system with proxy configuration. You add a proxy system and you have an external identity management system (such as, SAP Identity Management) in place. For more information, see: [Proxy Systems \[page 981\]](#).
3. Configure the connection details for your systems if they are not automatically set in your bundle tenant. You have the following options:
    - Add properties in the Identity Provisioning UI and provide the required connection information. For more information, refer to the respective provisioning systems (connectors) listed under [Supported Systems \[page 452\]](#) section where mandatory properties are specified.
    - Create a destination in your subaccount in SAP BTP cockpit and select it for the given provisioning system in the Identity Provisioning UI. If your bundle tenant is running on SAP BTP, Neo environment, the Identity Provisioning admin user must have the [Manage Destinations](#) role assigned. For more information, see [Manage Authorizations in Neo Environment → Bundle Tenants \[page 1490\]](#).

### → Recommendation

Creating a destination is mandatory for configuring SAP Application Server ABAP provisioning systems and on-premise systems using the Cloud Connector for which a [Location ID](#) is configured.

You can also use it if you need to reuse one and the same configuration for multiple provisioning systems. In all other cases, we recommend that you use the [Properties](#) tab.

If your bundle tenant is running on the infrastructure of SAP Cloud Identity Services, connectivity destinations can only be created for SAP Application Server ABAP provisioning systems. For more information, see [SAP Application Server ABAP \[page 484\]](#).

4. Define what data you want to provision. You have the following options:
  - Adapt the default transformation logic or use it as-is. For more information, see: [Transformations \[page 323\]](#).
  - Configure filtering properties for users and groups. For more information, see: [Properties \[page 90\]](#).
5. Run a provisioning job manually or set a time interval for scheduled jobs. For more information, see: [Start and Stop Provisioning Jobs \[page 1524\]](#).

## Related Information

[Bundle Tenants and Connectors \[page 422\]](#)

### 1.4.1 Obtain a Bundle Tenant

When an SAP cloud solution bundles with SAP Cloud Identity Services, you are entitled to receive Identity Authentication and Identity Provisioning tenants without additional costs on the purchase of the corresponding SAP cloud solution's license. These Identity Authentication and Identity Provisioning tenants come preconfigured with the SAP cloud solution.

## Context

Below you can find detailed information about Identity Provisioning bundle tenant and all available bundle options. The documentation distinguishes between bundle tenant and bundle option as follows:

A **bundle tenant** is an instance of Identity Provisioning that comes with a set of preconfigured provisioning systems relevant to one or more bundled SAP cloud solutions.

A **bundle option** is an SAP cloud solution bundled with Identity Provisioning and Identity Authentication that comes with a set of provisioning systems relevant to this SAP cloud solution.

Question	Answer
How do I obtain a bundle tenant?	<p>The way you obtain a bundle tenant depends on the SAP cloud solution you have purchased. It can be an automatic process or a manual one (by creating an incident).</p> <ul style="list-style-type: none"> <li> <p>Access tenant URLs sent by emails</p> <p>After purchasing an SAP cloud solution, you receive an onboarding email from SAP for each bundle tenant (test and productive) that is provisioned to you. The email contains a URL link which you can use to directly access the Identity Provisioning UI. For example, this applies to <a href="#">SAP Commissions</a>. For more information, see <a href="#">SAP Commissions Bundle [page 430]</a>.</p> </li> <li> <p>Trigger tenant creation</p> <p>After purchasing an SAP cloud solution, you can obtain Identity Provisioning bundle tenant by triggering the tenant creation. For example, this applies to <a href="#">SAP SuccessFactors</a>, where to obtain a bundle tenant with Identity Authentication and Identity Provisioning, you need to complete an upgrade process in <b>SAP SuccessFactors Upgrade Center</b>. For more information, see <a href="#">SAP SuccessFactors Bundle [page 438]</a>.</p> </li> <li> <p>Create a subscription in SAP BTP cockpit</p> <p>After purchasing a global account for SAP Business Technology Platform (in short, SAP BTP), you obtain Identity Provisioning tenant by subscribing your subaccount to the <a href="#">Cloud Identity Services</a> application in SAP BTP cockpit. This applies to <a href="#">SAP BTP</a>. For more information, see <a href="#">SAP Business Technology Platform Bundle [page 427]</a>.</p> </li> <li> <p>Create an incident</p> <p>After purchasing an SAP cloud solution, you can obtain Identity Provisioning bundle tenant by creating an incident. For example, this applies to <a href="#">SAP Cloud Identity Access Governance</a>. For more information, see <a href="#">SAP Cloud Identity Access Governance Bundle [page 431]</a>.</p> </li> </ul>



Question	Answer
<b>How many provisioning systems are available in my bundle tenant?</b>	<p>When an SAP cloud solution is bundled with Identity Provisioning and Identity Authentication, you get a bundle tenant with a set of provisioning systems (source, target and proxy) for which you have a license. Those systems are preconfigured in your tenant.</p> <p>Further usage of Identity Provisioning connectors and their availability depend on the infrastructure/environment your bundle tenant is running on. For more information, see <a href="#">Bundle Tenants and Connectors [page 422]</a></p>
<b>What happens when I purchase more SAP cloud solutions?</b>	<p>If you purchase more than one SAP cloud solutions, the scope of your first bundle will be extended. You will not get additional tenants.</p> <p>For example, if you purchase SAP Business Technology Platform first and you obtain your Identity Provisioning bundle tenant on Neo environment, this tenant will have all SAP BTP related provisioning systems. Later, if you purchase <a href="#">SAP SuccessFactors</a>, your bundle tenant will be extended with SAP SuccessFactors and its relevant systems.</p>
<b>Can I obtain additional bundle tenants?</b>	<p>When you purchase an SAP cloud solution, that is bundled with SAP Cloud Identity Services, you are entitled to two Identity Provisioning bundle tenants – one for testing and one for productive purposes. Regardless of how many SAP cloud solutions you have purchased, you still get two tenants.</p> <p>For more information on how to get additional tenant, see <a href="#">Getting an Additional Tenant</a></p> <p>By default, additional tenant is always created with SAP Business Technology Platform bundle option. If you have valid licenses to any other solution that bundles Identity Provisioning, those provisioning systems will also be enabled.</p> <p>As a result, once you get the additional Identity Provisioning tenant, you will have all provisioning systems enabled for your first bundled SAP cloud solution along with the provisioning systems relevant for your SAP BTP bundle.</p>
<b>How is my bundle tenant updated?</b>	<p>When a new provisioning system is added to a given bundle option, this provisioning system becomes available for every bundle tenant containing this bundle option.</p> <p>When a new version of a provisioning system is released, each bundle tenant containing the provisioning system is automatically updated with the new version of the given system.</p>

## Question

## Answer

**How do I proceed after obtaining a bundle tenant?**

Depending on how you obtain your bundle tenant - manually or automatically, you can expect the following:


- Manually obtained bundle tenants - your source and target systems are not configured. That is, you need to set up the connection to the relevant source and target systems.
- Automatically obtained bundle tenants - your source and target systems are preconfigured. That is, the connection to the relevant source and target systems are already set up and you can run provisioning jobs.

For some bundle tenants, such as [SAP Jam Collaboration](#), when the two tenants are created for you, the first initial [Read Job](#) is scheduled and starts almost immediately. When the job is finished, all users from the source system are provisioned to the target one. The credentials that are used during this initial job are a predefined user and an initial dummy password, which you later must replace with your own.


## Question

Can I view all SAP Cloud Identity Services tenants that are assigned to my customer ID and tenant administrators?

## Answer

As an SAP customer, you can view all your Identity Authentication and Identity Provisioning tenants by accessing the *SAP Cloud Identity Services - Tenants* application at the following URL: <https://iamtenants.accounts.cloud.sap/>. It displays the type of the tenant (test or productive), the date it was created, the region where it is available and the tenant administrators.

For more information about the data you can view and how to log on, see:

- [Viewing Assigned Tenants and Administrators](#)
- [New! Check on one single page all of your Identity Authentication and Identity Provisioning tenants and administrators](#)

### i Note

When the Identity Provisioning tenant is initially provisioned to your organization, only one user is added as a tenant administrator. After that, due to possible legal and security issues, SAP adds additional tenant administrators only in exceptional cases (for example, the existing administrator left the company, or for some reason there is no active administrator for this tenant).

To avoid access-related issues in such cases, it is always a good practice for you to assign more than one administrators. Adding additional ones is exclusively in the responsibility of the current tenant administrators. For more information, see [Manage Authorizations \[page 1487\]](#).

For more information about all available bundle options, see [Bundle Tenants and Connectors \[page 422\]](#)

## 1.4.2 Getting a Trial Tenant

You can create an SAP Cloud Identity Services trial tenant from an SAP BTP trial account.

### Prerequisites

You have a free account on SAP BTP Trial, as described in [Get a Free Account on SAP BTP Trial](#)

## Context

A trial tenant is intended for testing purposes. It allows you to try out and explore SAP Cloud Identity Services – Identity Authentication and Identity Provisioning. It is free of charge and limited to a trial period in accordance with the [SAP BTP trial terms and conditions](#).

A trial tenant differs from a test tenant. Trial tenants can be used for a limited period, while test tenants are provided to customers in addition to their productive tenants where they can test new features before implementing them productively.

### ! Restriction

- You can have only one trial tenant per SAP BTP global account regardless of how many subaccounts you create. If you have multiple subaccounts, and you've got trial tenants for them, the subaccounts will use one and the same trial tenant.
- You can have up to 50 users in a trial tenant.
- You can't connect to on-premise systems using the cloud connector. It's not possible to subscribe to the Cloud Identity Services connectivity plan in your trial subaccount.

## Procedure

1. Go to the [SAP BTP Trial page](#) and click [Log On](#).
2. Choose [Go to Your Trial Account](#) and open your subaccount.
3. In the navigation area, choose [Entitlements](#) and then ► [Configure Entitlements](#) ► [Add Service Plans](#) ►.
4. Select [Cloud Identity Services](#) from the list of entitlements available for this subaccount.
5. Mark the [default \(Application\)](#) plan. The 0 number in the [Add 0 Service Plan](#) button increments to 1.
6. Choose the [Add 1 Service Plan](#) button.
7. Choose [Save](#).
8. In the navigation area, choose ► [Services](#) ► [Instances and Subscriptions](#) ►.
9. Choose [Create](#) in the top-right corner.

A wizard opens, offering you to configure your instance. Provide the following information:

1. Choose [Cloud Identity Services](#).
2. Select the [default](#) plan.
3. Choose [Create](#).

## Results

You will receive an email with subject *Activate Your Account for Identity Authentication Service*. Follow the instructions in the email to activate your account for administration console of SAP Cloud Identity Services (formerly known as administration console of Identity Authentication).

After a successful login, you open the SAP Cloud Identity Services administration console. You are the initial administrator user, listed under ► [Users & Authorizations](#) ► [Administrators](#) ►.

## Next Steps

You can now start testing Identity Authentication and Identity Provisioning features. For more information, refer to:

- [Identity Authentication Operations](#)
- [Identity Provisioning Operations \[page 1476\]](#)

### i Note

You can delete your trial tenant by deleting the subaccount or by deleting the Cloud Identity Services application under the Subscriptions section. For more information, see [Delete a Subaccount](#).

### i Note

If you encounter any issues and you need support, send an email to [SAPCPTrialSupport@sap.com](mailto:SAPCPTrialSupport@sap.com) or start a discussion in [SAP Community](#) 🗨️.

## Related Information

[Trial Accounts and Free Tier](#)

**Blog Post:** [SAP Cloud Identity Services offered as Trial Version](#) 📄

## 1.4.3 Access Identity Provisioning UI of Bundle Tenants

Access the Identity Provisioning UI when the service is bundled as part of an SAP cloud solution's license.

## Prerequisites

- You have purchased an SAP cloud solution that bundles Identity Provisioning and Identity Authentication. This solution is automatically available in the Identity Provisioning UI as a source and/or target system, that is, as a system to read entities from and/or a system to write entities to. User stores, such as Microsoft Azure Active Directory, are available as source (read-only) systems for integration purposes. See: [Obtain a Bundle Tenant \[page 407\]](#)

## Context

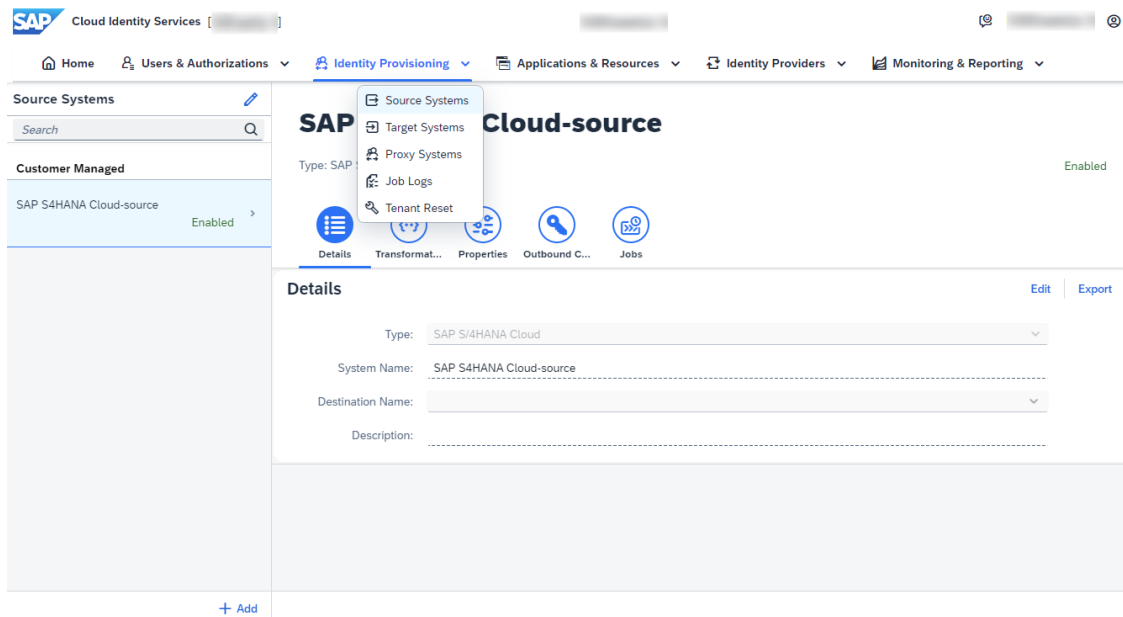
To access the Identity Provisioning UI, you need to open the URL link that you received when obtaining your bundle tenant. You get the URLs for your test and productive tenant in the onboarding e-mails from SAP. If you obtained it manually (by opening an incident), you get the URLs from the incident.

### → Remember

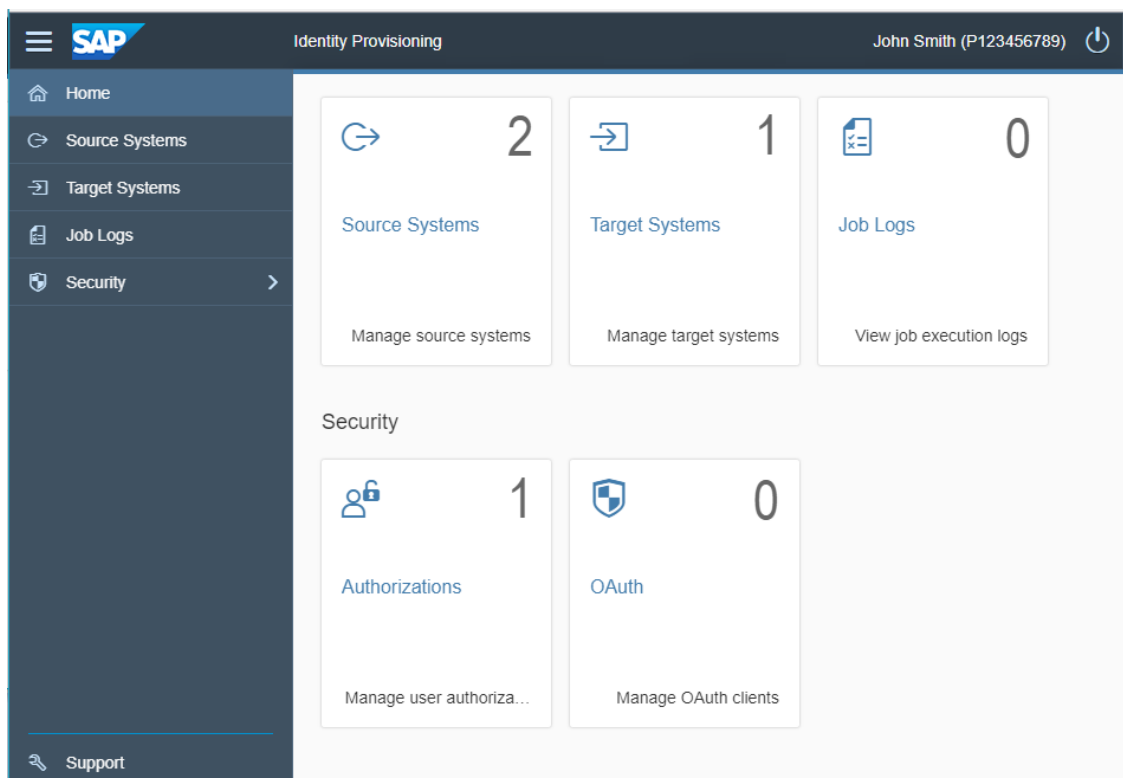
When you receive the onboarding e-mails from SAP and access the Identity Provisioning UI for the first time, your source and target systems will be preconfigured. That is, the connection to the relevant source and target systems will be set up and you can run provisioning jobs. For some bundle tenants, such as SAP Jam Collaboration, the first initial [Read Job](#) is scheduled and starts almost immediately.

## Procedure

1. Open the test or productive URL you received either from the contract e-mails or from the incident you opened. Depending on the infrastructure/environment your bundle tenant runs on, the URL follows the pattern:
  - **SAP Cloud Identity infrastructure:** `https://<ias-host>/admin`, where the Identity Provisioning tenant URL uses the host of the corresponding Identity Authentication tenant of the customer. If your tenant URL is `https://<ias-host>/ips`, you are redirected to `https://<ias-host>/admin`. This opens the common administration console of SAP Cloud Identity Services, where the provisioning functionality is embedded under [Identity Provisioning](#) section.
  - **Neo environment:** `https://ips-<consumer_account>.dispatcher.<region_host>/webapp/index.html`  
This will open the administration console of Identity Provisioning.
2. Log on to Identity Provisioning with your administration credentials.
  - **SAP Cloud Identity infrastructure:** Identity Provisioning administrator authenticates to SAP Cloud Identity Services administration console. The login credentials are set when the initial administrator logs in for the first time. The admin user must have the [Manage Identity Provisioning](#) role enabled in the SAP Cloud Identity Services administration console.  
For more information, see [Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#).
  - **Neo environment:** Using his or her S-user credentials, Identity Provisioning administrator authenticates to the admin console of the service, that is provided by SAP in the welcoming onboarding email. The admin user has the [Manage Identity Provisioning](#) role enabled in the Identity Provisioning admin console.  
For more information, see [Manage Authorizations in Neo Environment \[page 1490\]](#).
3. Navigate through Identity Provisioning.
  - **SAP Cloud Identity infrastructure:** Your starting point is the list of [Source Systems](#) in the administration console of SAP Cloud Identity Services. The entire provisioning functionality can be accessed through the navigation area under [Identity Provisioning](#) → [Source Systems](#), [Target Systems](#), [Proxy Systems](#), [Job Logs](#) and [Tenant Reset](#).



- **Neo environment:**  
Your starting point is the [Home](#) page of the Identity Provisioning administration console along with the following tiles:  
[Source Systems](#), [Target Systems](#), [Proxy Systems](#), [Job Logs](#), [Authorizations](#), and [OAuth](#).



### Note

In both standalone and bundle cases, secure communication is provided between this HTML5 application and the SAP BTP cockpit, realized by principal propagation. Unlike the standalone case

however, with your bundle account you obtain the Identity Provisioning as *software as a service*. That means, we provide you with a global SAP Business Technology Platform account, and you don't need to operate in the platform cockpit.

All source and target systems in your bundle tenant are available as proxy systems.

## Next Steps

1. Create additional administrator users.

### i Note

When the Identity Provisioning tenant is initially provisioned to your organization, only one user is added as a tenant administrator. After that, due to possible legal and security issues, SAP adds additional tenant administrators only in exceptional cases (for example, the existing administrator left the company, or for some reason there is no active administrator for this tenant).

To avoid access-related issues in such cases, it is always a good practice for you to assign more than one administrators. Adding additional ones is exclusively in the responsibility of the current tenant administrators. For more information, see [Manage Authorizations \[page 1487\]](#).

2. Continue with configuring your [Bundle Tenants and Connectors \[page 422\]](#).

In case of issues when accessing your Identity Provisioning bundle tenant, open an incident to **BC-IAM-IPS** component. Alternatively, you can ask questions in the SAP Community. For more information, see [Getting Support \[page 1620\]](#).

## 1.4.4 Provisioning Systems for Bundle Tenants

Provisioning systems (connectors) availability in bundle tenants depends on the infrastructure/environment your tenants are running on.

### i Note

Bundle tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, which are obtained after purchasing the following SAP cloud solutions, are the least restrictive in terms of connectors availability:

- SAP Cloud Identity Access Governance bundle tenant has all connectors enabled.
- SAP Jam Collaboration bundle tenant has all source and proxy connectors enabled.

## SAP Cloud Identity Infrastructure

If your bundle tenant is running on the infrastructure of SAP Cloud Identity Services, most of the connectors are enabled by default, with few exceptions described in [Bundle Tenants and Connectors \[page 422\]](#).




## SAP BTP, Neo Environment

If your bundle tenant is running on SAP BTP, Neo environment, a limited number of connectors are enabled by default. The table below lists the enabled provisioning systems (connectors) and the SAP cloud solution bundle they are relevant for:

SAP Cloud Solution	Provisioning Systems	System Type
SAP Business Technology Platform Bundle [page 427]	Identity Authentication	Source, Target, Proxy
	SAP BTP ABAP Environment	Source, Target, Proxy
	SAP BTP Account Members (Neo)	Source, Target, Proxy
	SAP BTP Java/HTML5 apps (Neo)	Source, Target, Proxy
	SAP BTP XS Advanced UAA (Cloud Foundry)	Source, Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	Microsoft Azure Active Directory	Source, Proxy
	Microsoft Active Directory	Source, Proxy
	Google G Suite	Source, Proxy
	SCIM System	Source, Proxy
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
SAP Build Work Zone, advanced edition Bundle [page 444]	Identity Authentication	Source, Target, Proxy
	SAP Build Work Zone, advanced edition	Source, Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	Microsoft Azure Active Directory	Source, Proxy
	Microsoft Active Directory	Source, Proxy
	Google G Suite	Source, Proxy
	SAP BTP XS Advanced UAA (Cloud Foundry)	Target, Proxy

SAP Cloud Solution	Provisioning Systems	System Type
	SAP Build Work Zone, standard edition	Target, Proxy
	SCIM System	Source, Proxy
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
SAP Commissions Bundle <a href="#">[page 430]</a>	Identity Authentication	Source, Target, Proxy
	SAP Commissions	Source, Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	SAP Analytics Cloud	Target, Proxy
	Microsoft Azure Active Directory	Source, Proxy
	Microsoft Active Directory	Source, Proxy
	Google G Suite	Source, Proxy
	SCIM System	Source, Proxy
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
SAP Cloud Identity Access Governance Bundle <a href="#">[page 431]</a>	<a href="#">All system connectors [page 452]</a>	Source, Target, Proxy
SAP Integrated Business Planning for Supply Chain Bundle <a href="#">[page 432]</a>	Identity Authentication	Source, Target, Proxy
	SAP Integrated Business Planning for Supply Chain	Source, Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	Microsoft Azure Active Directory	Source, Proxy
	Microsoft Active Directory	Source, Proxy
	Google G Suite	Source, Proxy
	SCIM System	Source, Proxy

SAP Cloud Solution	Provisioning Systems	System Type
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
SAP Jam Collaboration Bundle <a href="#">[page 435]</a>	Identity Authentication	Source, Target, Proxy
	SAP Jam Collaboration	Source, Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	All <a href="#">source [page 452]</a> and <a href="#">proxy [page 981]</a> system connectors	Source, Proxy
SAP Marketing Cloud Bundle <a href="#">[page 437]</a>	Identity Authentication	Source, Target, Proxy
	SAP Marketing Cloud	Source, Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	SAP Analytics Cloud	Target, Proxy
	Microsoft Azure Active Directory	Source, Proxy
	Microsoft Active Directory	Source, Proxy
	Google G Suite	Source, Proxy
	SCIM System	Source, Proxy
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
SAP S/4HANA Cloud Bundle <a href="#">[page 442]</a> (Only applicable for <a href="#">SAP S/4HANA Cloud Essentials Edition</a>  )	Identity Authentication	Source, Target, Proxy
	SAP S/4HANA Cloud	Source, Target, Proxy
	SAP Central Business Configuration	Source, Target, Proxy

SAP Cloud Solution	Provisioning Systems	System Type
	SCIM System	Source, <b>Target*</b> , Proxy
	<div> <b>! Restriction</b>            *SCIM System can be used as a target system only in provisioning scenarios with SAP Central Business Configuration.         </div>	
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	SAP Analytics Cloud	Target, Proxy
	Microsoft Azure Active Directory	Source, Proxy
	Microsoft Active Directory	Source, Proxy
	Google G Suite	Source, Proxy
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
SAP SuccessFactors Bundle [page 438]	Identity Authentication	Source, Target, Proxy
	SAP SuccessFactors	Source, Proxy
	SAP Analytics Cloud	Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	SCIM System	Source, Proxy
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
SAP SuccessFactors Learning Bundle [page 440]	Identity Authentication	Target, Proxy
	SAP SuccessFactors Learning	Source, Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	SCIM System	Source, Proxy

SAP Cloud Solution	Provisioning Systems	System Type
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
<a href="#">Sales Cloud – Analytics &amp; AI Bundle [page 446]</a>	Identity Authentication	Source, Proxy
	Sales Cloud – Analytics & AI	Target, Proxy
	SAP Analytics Cloud	Target, Proxy
	Microsoft Azure Active Directory	Source, Proxy
	Microsoft Active Directory	Source, Proxy
	Google G Suite	Source, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	SCIM System	Source, Proxy
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source
<a href="#">SAP Fieldglass Bundle [page 448]</a>	Identity Authentication	Source, Target, Proxy
	SAP Fieldglass	Source, Target, Proxy
	SAP Application Server ABAP	Source, Target, Proxy
	SAP S/4HANA On-Premise	Source, Target, Proxy
	Microsoft Azure Active Directory	Source, Proxy
	Microsoft Active Directory	Source, Proxy
	Google G Suite	Source, Proxy
	SCIM System	Source, Proxy
	LDAP Server	Source, Proxy
	SAP Enterprise Portal	Source

## Related Information

[Obtain a Bundle Tenant \[page 407\]](#)

## 1.5 Bundle Tenants and Connectors

A bundle tenant is an instance of Identity Provisioning that comes with a set of preconfigured provisioning systems (connectors) relevant to one or more bundled SAP cloud solutions.

The communication between Identity Provisioning and the preconfigured systems is automatically set up, therefore administrators are ready to run or schedule provisioning jobs. Further usage of provisioning systems (connectors) and their availability depend on the infrastructure/environment your bundle tenant is running on.

### i Note

Bundle tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment, which are obtained after purchasing the following SAP cloud solutions, are the least restrictive in terms of connectors availability:

- SAP Cloud Identity Access Governance bundle tenant has all connectors enabled.
- SAP Jam Collaboration bundle tenant has all source and proxy connectors enabled.

### Bundle tenants on SAP Cloud Identity infrastructure

If your bundle tenant is running on the infrastructure of SAP Cloud Identity Services, most of the connectors are enabled by default, with few exceptions described in *Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure* section below.

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only.

### Bundle tenants on SAP BTP, Neo environment

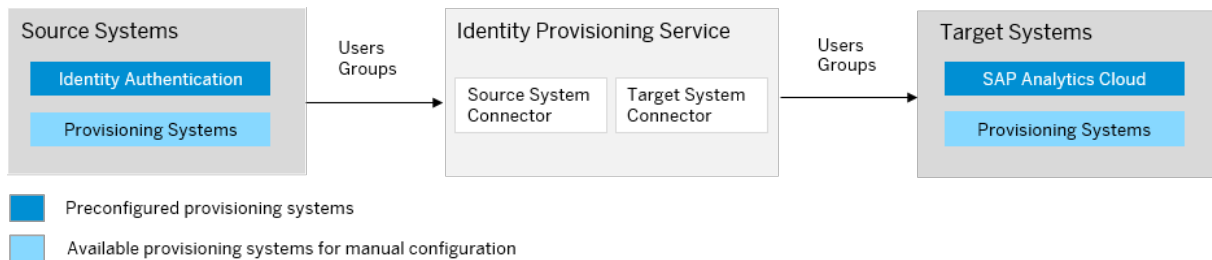
If your bundle tenant is running on SAP BTP, Neo environment, a limited number of connectors are enabled by default. You can extend the number of connectors by migrating the tenant to the SAP Cloud Identity infrastructure or by purchasing more SAP cloud solutions that bundle Identity Provisioning. In the latter case, along with the enabled provisioning systems of your bundle tenant, you will also get the provisioning systems relevant for the newly purchased solutions.

For more information, see [Migrate Identity Provisioning Bundle Tenant \[page 1536\]](#) and [Tenant Infrastructure \[page 10\]](#).

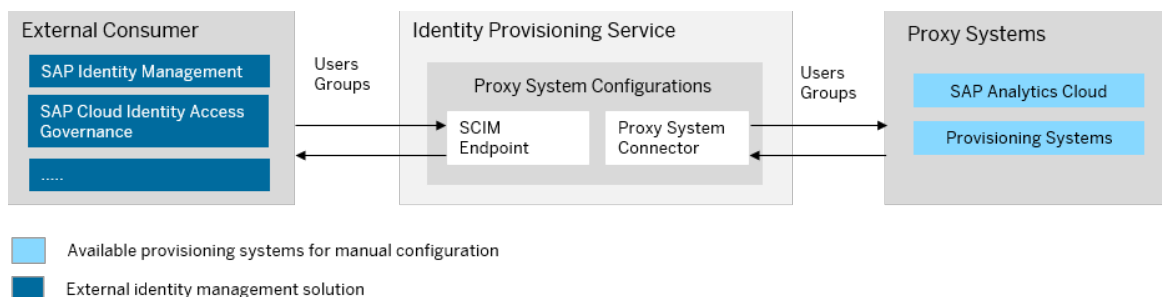
## How to Use Bundle Tenants

This example scenario explains the usage of Identity Provisioning bundle tenant that comes with a set of preconfigured systems relevant for a given SAP cloud solution. Although details vary from one bundle tenant to another, the overall process is the same.

The diagram below illustrates the expected provisioning flow from source to target systems (the default provisioning mode).



- After purchasing an SAP cloud solution that bundles Identity Provisioning, the technical contact person in your organization receives the welcome e-mail from SAP. He or she is granted the Administrator permissions for the bundle tenant and performs the initial logon by accessing the tenant URL provided in the e-mail. For more information, see [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#). The initial administrator is assigned the [Manage Identity Provisioning](#) role. He or she can configure provisioning systems, run jobs, view job logs, and add other users as administrators of the tenant. For more information, see: [Manage Authorizations \[page 1487\]](#).
- When opening the Identity Provisioning UI, the preconfigured systems are displayed in the respective *Source Systems* and *Target Systems* tile.
  - On the *Details* tab, the preconfigured systems are populated with their specific system name. For example:
    - `<System_name> - source` (Identity Authentication)
    - `<System_name> - target` (SAP Analytics Cloud)
  - On the *Properties* tab, the communication between Identity Provisioning and those systems is automatically set up. Connection properties are populated. The authentication type is specified. You have the following options: use [BasicAuthentication](#) or [ClientCertificateAuthentication](#), as described in [Manage Certificates \[page 1506\]](#). You can modify properties to further control how user data is transferred. For example:
    - You can configure filtering on source systems to provision users and groups matching specific criteria or to test the provisioning with few users and groups before you replicate all of them.
    - You can enable bulk operations on the target system to speed up the provisioning.
 For more information, see [List of Properties \[page 94\]](#), where you need to search for the properties with the relevant prefix of your system, like: `<ias>.user.filter` for Identity Authentication or `<sac>.support.bulk.operation` for SAP Analytics Cloud.
  - On the *Transformations* tab, the user and group resource mappings are displayed. Transformations of the preconfigured systems might be adapted for the scenarios relevant to your bundled SAP cloud solution. For more information on the various options you have to modify transformations, see [Transformations \[page 323\]](#).
- Administrators can run [Read](#) or [Resync](#) job for the preconfigured systems. For more information, see [Start and Stop Provisioning Jobs \[page 1524\]](#).
- Administrators can also manually add and configure source, target and proxy systems of their choice. Those systems are created with default properties and default transformations. For more information, follow the procedure described for your provisioning system under [Source Systems \[page 452\]](#) and [Target Systems \[page 702\]](#) sections. The diagram below illustrates the expected provisioning flow between an external identity management solution and a system with proxy configuration (the proxy provisioning mode). For more information, see [Proxy Systems \[page 981\]](#).



In this mode, Identity Provisioning is used for synchronizing user data to and from a central identity management solution (for example, the on-premise SAP Identity Management) and a provisioning system with proxy configuration (for example, SAP Analytics Cloud, embedded edition). Here, Identity Provisioning acts as a proxy between the identity management solution and the system with proxy configuration.

## Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure

Almost all connectors are available. The table below lists only the few provisioning systems which are not supported for bundle tenants running on SAP Cloud Identity infrastructure:

Availability	Provisioning Systems (Connectors)
Not supported as source, target and proxy systems	Cloud Foundry UAA Server
Not supported as target systems	SCIM
	LDAP Server
	Microsoft Active Directory
	Microsoft Azure Active Directory
	Google G suite

## Connectors Availability in Bundle Tenants on SAP BTP, Neo Environment

The table below lists the provisioning systems supported for bundle tenants running on SAP BTP, Neo environment. Some of them are available in all bundle tenants (for example, Identity Authentication), others are specific for a given bundle tenant (for example, SAP Fieldglass).

Also, some of these systems are SAP cloud solutions that bundle Identity Provisioning, and therefore are defined as bundle options themselves with their own set of relevant provisioning systems. For more information, refer to the bundle options listed in this section.



Connector	Source System	Target System	Proxy System
<i>Identity Authentication</i>	✓	✓	✓
<i>SAP Analytics Cloud</i>		✓	✓
<i>SAP Application Server ABAP</i>	✓	✓	✓
<i>SAP BTP ABAP environment</i>	✓	✓	✓
<i>SAP BTP Account Members (Neo)</i>	✓	✓	✓
<i>SAP BTP Java/HTML5 apps (Neo)</i>	✓	✓	✓
<i>SAP BTP XS Advanced UAA (Cloud Foundry)</i>	✓	✓	✓
<i>SAP Build Work Zone, advanced edition</i>	✓	✓	✓
<i>SAP Build Work Zone, standard edition</i>		✓	✓
<i>SAP Central Business Configuration</i>	✓	✓	✓
<i>SAP Commissions</i>	✓	✓	✓
<i>SAP Enterprise Portal</i>	✓		
<i>SAP Fieldglass</i>	✓	✓	✓
<i>SAP Integrated Business Planning for Supply Chain</i>	✓	✓	✓
<i>SAP Jam Collaboration</i>	✓	✓	✓
<i>SAP Marketing Cloud</i>	✓	✓	✓
<i>SAP S/4HANA Cloud</i>	✓	✓	✓
<i>SAP S/4HANA On-Premise</i>	✓	✓	✓
<i>SAP SuccessFactors</i>	✓		✓

Connector	Source System	Target System	Proxy System
<i>SAP SuccessFactors Learning</i>	✓	✓	✓
<i>Sales Cloud – Analytics &amp; AI</i>		✓	✓
<i>Google G Suite</i>	✓		✓
<i>Microsoft Active Directory</i>	✓		✓
<i>Microsoft Azure Active Directory</i>	✓		✓
<i>SCIM System</i>	✓		✓
<i>LDAP Server</i>	✓		✓

The table below lists the provisioning systems available in all bundle tenants and their use case.

Connector	Use Case
Identity Authentication	<p>Identity Authentication is available as source, target and proxy system in all bundle tenants, except for a target system in Sales Cloud – Analytics &amp; AI Bundle bundle).</p> <p>Using Identity Authentication as a source system is relevant for new customers of SAP cloud solutions (<b>green-field approach</b>). Users are created in Identity Authentication (self-registered, uploaded from files or provisioned via Identity Provisioning from another source system) and afterwards replicated to target systems.</p> <p>Using Identity Authentication as a target system is relevant for existing customers of SAP cloud solutions (<b>brown-field approach</b>). Users are already created in the SAP cloud solutions and afterwards provisioned to Identity Authentication which becomes the leading system in user provisioning.</p>
Microsoft Azure Active Directory	<p>These systems are available as source systems in all bundle tenants.</p> <p>The reason is that they are normally used as corporate user stores and therefore can act as central place for user management in customers landscapes.</p> <p>The only exception is SAP SuccessFactors bundle, where Microsoft Azure AD, Microsoft AD and Google G Suite are not available as source systems. However, its scope can be</p>
Microsoft Active Directory	
Google G Suite	
SAP Application Server ABAP	
SAP S/4HANA On-Premise	

Connector	Use Case
SCIM System	extended, and those systems can be enabled for reading entities upon purchasing more bundled SAP cloud solutions.
LDAP Server	<p>In addition to being available as source systems, SAP AS ABAP and SAP S/4HANA On-Premise are also available as target systems in all bundle tenants.</p> <p>The reason is that they are normally part of the customers landscapes and therefore can be used for both - reading and replicating entities, when implementing user management and provisioning.</p>

## 1.5.1 SAP Business Technology Platform Bundle

SAP Business Technology Platform bundle allows you to use the Identity Provisioning service for synchronizing user data between source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

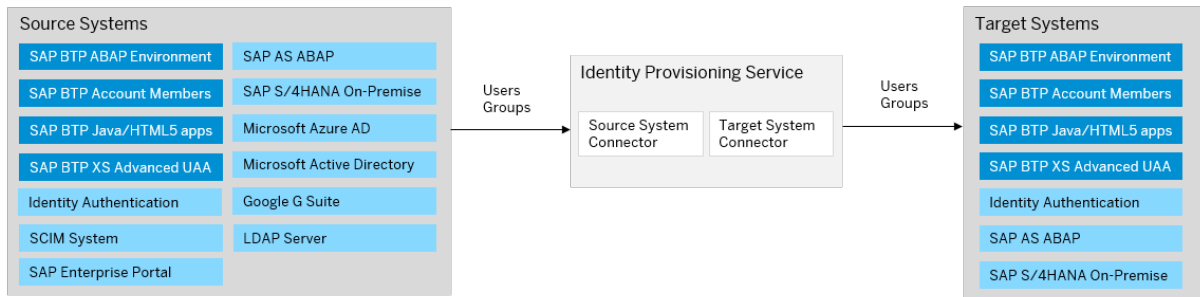
### i Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

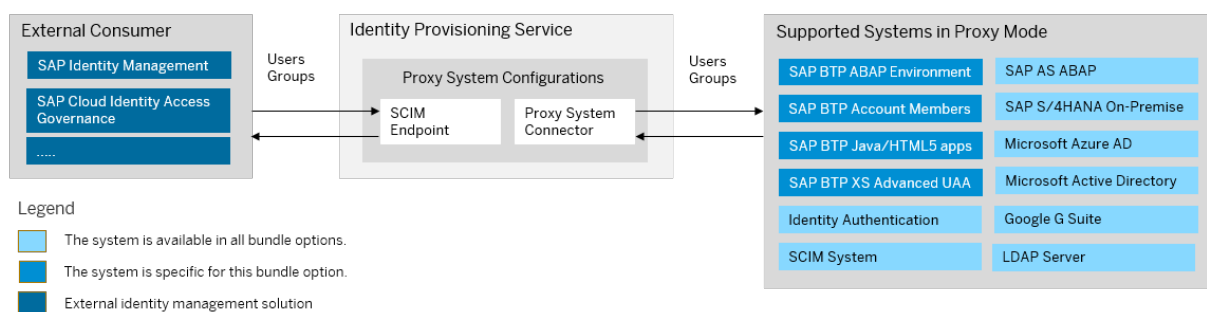
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

## Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



## How to Obtain

After purchasing a global account for SAP Business Technology Platform (in short, SAP BTP), you can obtain Identity Provisioning tenant. For this, your subaccount needs to be subscribed to the [Cloud Identity Services](#) application in SAP BTP cockpit. The application plan is currently available in the following Cloud Foundry regions:

- Australia (Sydney) - AWS
- Brazil (São Pauli) - AWS
- Canada (Montreal) - AWS
- Europe (Frankfurt) - AWS
- Europe (Netherlands) - AZURE
- Japan (Tokyo) - AWS
- Japan (Tokyo) - AZURE
- Singapore - AWS
- Singapore - AZURE
- US Central (IA) - GCP
- US East (VA) - AWS
- US East (VA) - AZURE
- US West (WA) - AZURE

## i Note

To obtain the Identity Provisioning tenant, make sure your subaccount is subscribed to the application in the correct region. If you do this in multiple regions, the tenant you obtain in the first region will be reused in all other regions.

Also, if you already have an existing Identity Provisioning, it will be reused for the relevant provisioning systems entitled to your SAP BTP product.

Follow the steps to activate your tenant:

1. In SAP BTP cockpit, choose your subaccount in the SAP BTP, Cloud Foundry environment.
2. Navigate to ► [Services](#) ► [Service Marketplace](#) ► and select [Cloud Identity Services](#).

## i Note

If you don't see **Cloud Identity Service**, navigate to [Entitlements](#) and add the default plan.

3. Under [Application Plans](#), choose [default](#) and then [Next](#).
4. In the [Cloud Service Type](#) dropdown, choose the tenant type. You have the following options:
  - [Test](#)
  - [Productive](#) - This is the default value.
5. Choose [Next](#) and then [Create](#) to make a subscription to this application, bound to your customer ID. The **default** application plan allows you to consume Identity Provisioning, Identity Authentication, and Identity Directory. The new test or productive tenant will be created only if there isn't an existing one already bound to your customer ID, regardless of the region. The default test and productive tenants will be created in the same region.
6. In the side menu of the SAP BTP cockpit, go to ► [Services](#) ► [Instances and Subscriptions](#) ►. Under the [Subscriptions](#) tab, in the table, you can find the [Cloud Identity Services](#) application.
7. Click the **...** ([Actions](#)) at the end of the subscription row and select [Go to Application](#). This opens the Identity Provisioning administration console. The URL of your Identity Provisioning bundle tenant follows the pattern:
  - `https://<ias-host>/ips`  
This is valid for new Identity Provisioning bundle tenants created after March 15, 2022. They run on SAP Cloud Identity infrastructure.
  - `https://ips-<consumer_account>.dispatcher.<region_host>/webapp/index.html`  
This is valid for existing (reused) Identity Provisioning tenants created before March 15, 2022. They run on SAP BTP, Neo environment.

## i Note

You can remove the subscription to the [Cloud Identity Services](#) application, that is to offboard your Identity Provisioning tenant, if you click the **...** ([Actions](#)) and select [Delete](#). Be aware of the following:

- If your Identity Provisioning tenant is obtained only for SAP BTP and no other SAP cloud solutions are reusing it, this tenant will be deleted immediately.
- If your Identity Provisioning tenant is reused in other SAP cloud solutions that bundle the service, this tenant will not be deleted until all SAP cloud solutions offboard it.

## Related Information

[Tenant Model and Licensing → Getting a Tenant](#)

<https://help.sap.com/viewer/product/BTP/Cloud/en-US>

## 1.5.2 SAP Commissions Bundle

SAP Commissions bundle allows you to use the Identity Provisioning service for synchronizing user data between source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

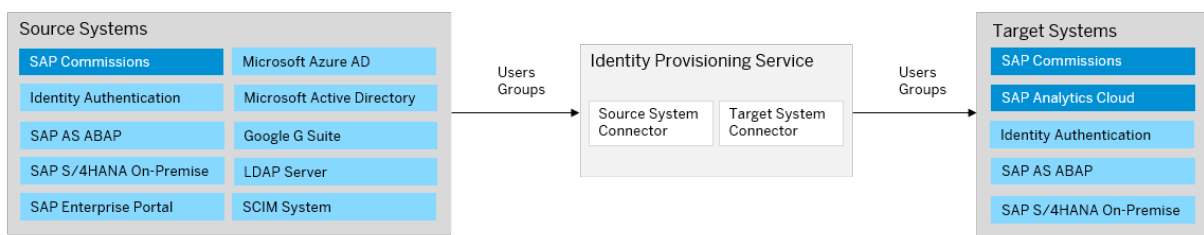
### Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

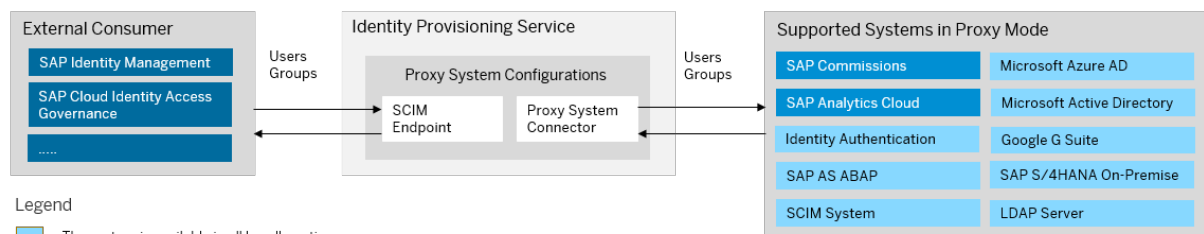
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

### Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



#### Legend

- The system is available in all bundle options.
- The system is specific for this bundle option.
- External identity management solution

## How to Obtain

After purchasing SAP Commissions, you'll receive two e-mails from SAP. According to your contract with SAP, a technical contact person has been chosen as the first user of the Identity Provisioning service, who is granted with *Administrator* permissions. In these e-mails from SAP, you'll find the ID of this administrator (their P- or S-user) and their e-mail address. They can access the Identity Provisioning UI with their user credentials.

Each e-mail from SAP contains also a URL link that you, as an administrator, can use to directly access the Identity Provisioning UI. These two URLs are related to two different Identity Provisioning tenants – the first one you can use for *testing* purposes, and the second one – for *productive* provisioning configurations and jobs.

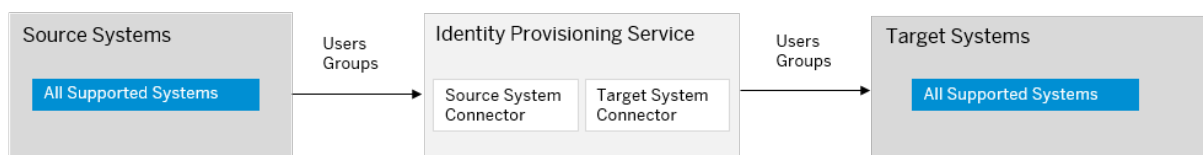
## Related Information

[SAP Commissions](#)

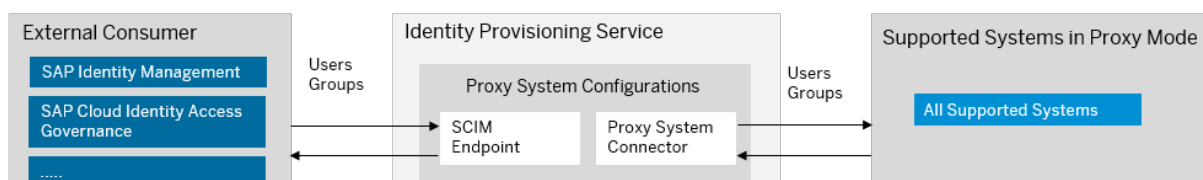
### 1.5.3 SAP Cloud Identity Access Governance Bundle



SAP SAP Cloud Identity Access Governance bundle allows you to use the Identity Provisioning service for synchronizing user data between all supported source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

**Default Mode:** Provision user data from source to target systems



**Proxy Mode:** Provision user data to and from central identity management solution and systems with proxy configuration



-  Supported systems for this bundle option
-  External identity management solution

## How to Obtain

After purchasing SAP Cloud Identity Access Governance, if your license includes only SAP Cloud Identity Access Governance, you can obtain the Identity Provisioning service as well.

### i Note

You won't be charged any extra fee as Identity Provisioning service has been officially integrated in the SAP Cloud Identity Access Governance license.

You already have Identity Authentication (a mandatory service), which enables you to sign in and authenticate in the Identity Provisioning UI and within SAP Business Technology Platform.

To obtain Identity Provisioning, you need to create an incident. Follow the steps:

1. Create an incident to component [GRC-IAG-OPS \(SAP Cloud Identity Access Governance\)](#).
2. Explain that you've purchased an SAP Cloud Identity Access Governance product and you require access to the Identity Provisioning service.
3. Specify the S-user to be assigned as the first administrator of the Identity Provisioning tenants. Later, this S-user can add other users as administrators.  
For complete information on what you need to specify in the incident, see [Connecting Identity Provisioning Tenant](#) in *SAP Cloud Identity Access Governance Admin Guide*

## Related Information

[SAP Cloud Identity Access Governance](#)

## 1.5.4 SAP Integrated Business Planning for Supply Chain Bundle

SAP Integrated Business Planning for Supply Chain (in short, SAP IBP) bundle allows you to use the Identity Provisioning service for synchronizing user data between source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

### i Note

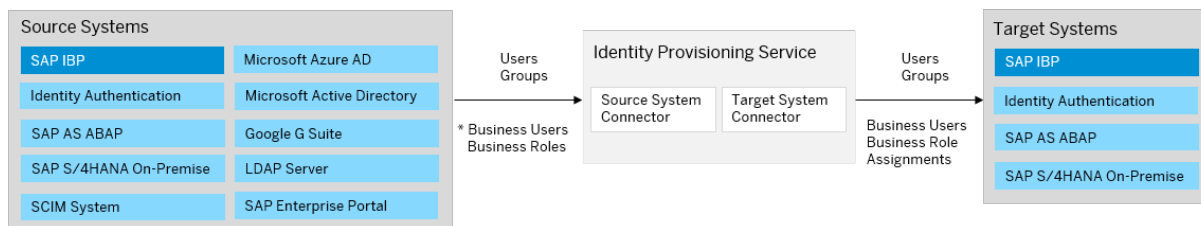
As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

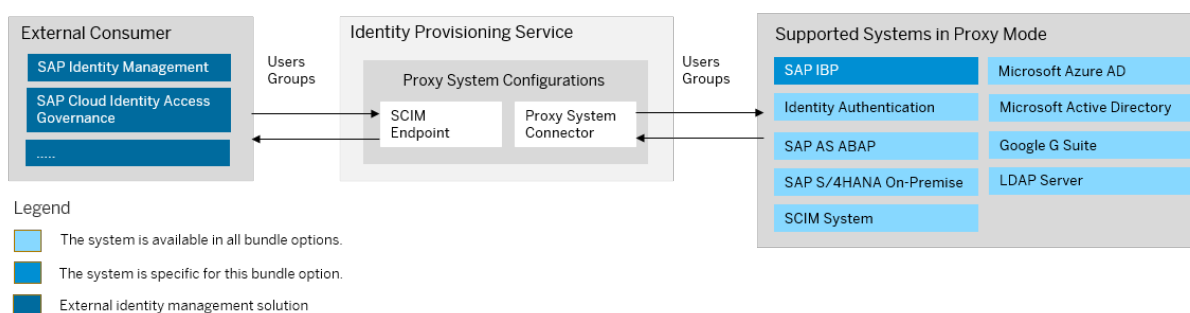


## Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



## How to Obtain

Since November 2022, the Identity Provisioning service is provided together with SAP IBP by default. For information about accessing your Identity Provisioning tenant as an administrator, see [Initial Administrator User](#). If your system has been provisioned before November 2022, you can obtain an Identity Provisioning bundle tenant by creating an incident as follows:

1. Create an incident to component *SCM-IBP-OPS-SRV (Cloud Operations Service Requests)*.
2. Specify the S-user to be assigned as the first administrator of the Identity Provisioning tenants. Later, this S-user can add other users as administrators.
3. Specify the URLs to your SAP IBP **Quality** and **Productive** systems.

In the reply of your incident, you'll receive two URLs related to two Identity Provisioning tenants.

- The first URL will be bound to your **Quality** instance, and you can use it for testing purposes.
- The second URL will be bound to your **Productive** instance, and you can use it for productive provisioning configurations and jobs. This bounding principle is applied to your Identity Authentication tenants as well.

If you encounter issues with accessing your Identity Provisioning UI, create an incident to component *BC-IAM-IPS*.

## How to Use

The scope of the SAP IBP bundle tenant includes the provisioning systems (connectors) displayed in the diagram above.

By default, the communication between SAP IBP and Identity Provisioning is automatically set up. Also, Identity Authentication is preconfigured as a source system and SAP IBP is preconfigured as a target system. This means that you can start running provisioning jobs manually or set a time interval for scheduled jobs. See: [Start and Stop Provisioning Jobs \[page 1524\]](#)

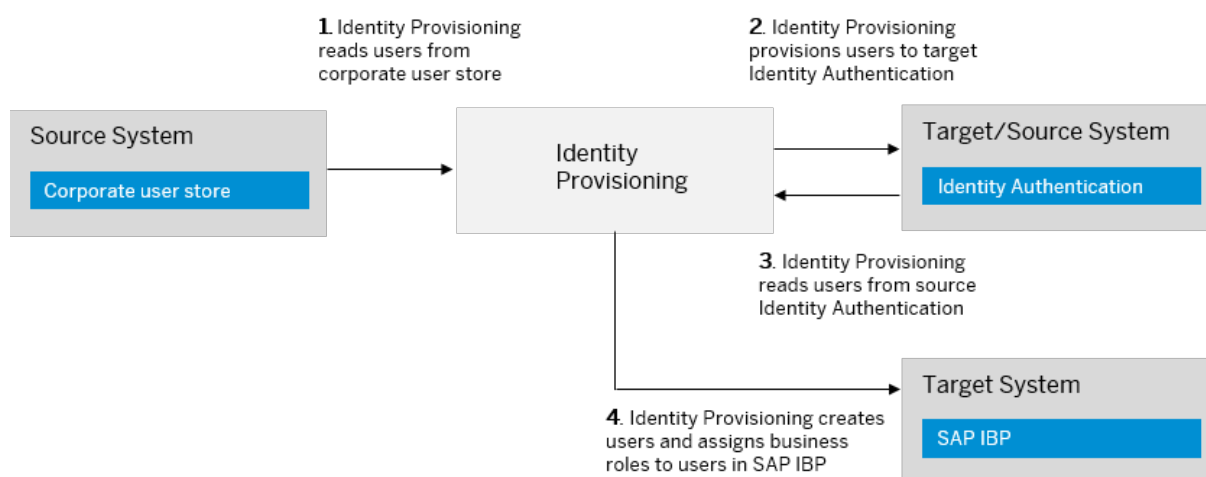
The other provisioning systems in the scope of this bundle are enabled. This means that you can start adding and configuring them in the Identity Provisioning UI. See: [Add a System \[page 1477\]](#)

### Note

Users and groups in Identity Authentication correspond to business users and business roles in SAP IBP. The user groups defined in SAP IBP are not considered by the Identity Provisioning.

### Example

In this example scenario, users are first provisioned to Identity Authentication (target system) from a corporate user store, such as Microsoft Azure AD (source system). Afterwards, those users are provisioned from Identity Authentication (this time set up as a source system) to SAP IBP (target system).



For more information about SAP IBP integration scenarios, see: [Integrating Identity Provisioning Service with SAP IBP](#)

For more information about the provisioning systems (source, target and proxy) relevant to SAP IBP bundle, see: [Supported Systems \[page 452\]](#)

## Related Information

[SAP Integrated Business Planning for Supply Chain](#)

## 1.5.5 SAP Jam Collaboration Bundle

SAP Jam Collaboration bundles with SAP Cloud Identity Services – Identity Authentication and Identity Provisioning.

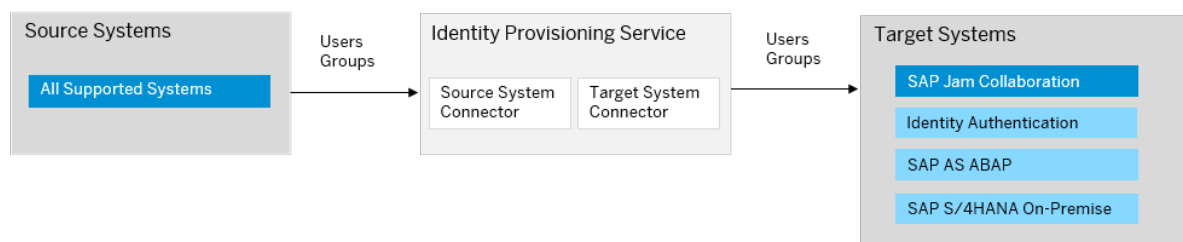
### Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

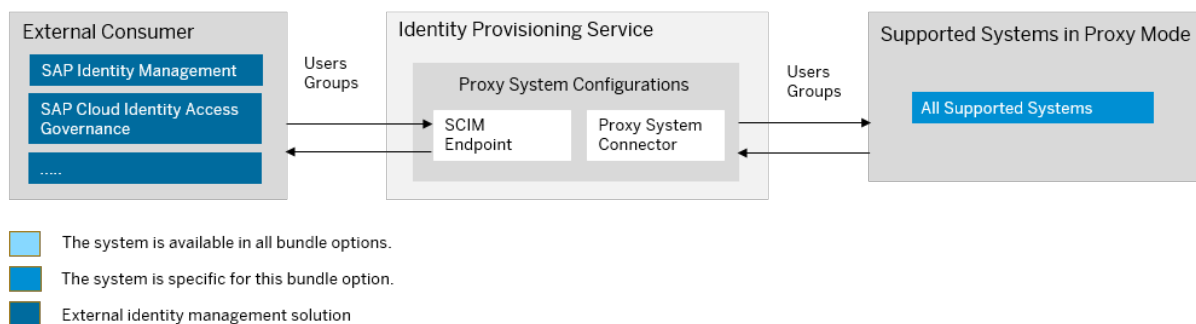
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

### Bundle Tenant on Neo Environment

**Default Mode:** Provision user data from source to target systems



**Proxy Mode:** Provision user data to and from central identity management solution and systems with proxy configuration



## How to Obtain

- Your license contains Identity Provisioning**

After purchasing SAP Jam Collaboration, you'll receive two e-mails from SAP. According to your contract with SAP, a technical contact person has been chosen as the first user of the Identity Provisioning service, who is granted with *Administrator* permissions. In these e-mails from SAP, you'll find the ID of this administrator (their P- or S-user) and their e-mail address. They can access the Identity Provisioning UI with their user credentials.

Each e-mail from SAP contains also a URL link that you, as an administrator, can use to directly access the Identity Provisioning UI. These two URLs are related to two different Identity Provisioning tenants – the first

one you can use for [testing](#) purposes, and the second one – for [productive](#) provisioning configurations and jobs.

If you encounter issues with accessing your Identity Provisioning UI, create an incident to component [BC-IAM-IPS](#).

- **Your license does not contain Identity Provisioning**

After the successful purchase, if your license includes only SAP Jam Collaboration, you can request a tenant for the Identity Provisioning service.

#### **i Note**

You won't be charged any extra fee as Identity Provisioning service has been officially integrated in the SAP Jam license.

See the following blog post: [SAP Jam now comes with the Identity Provisioning service](#) 

You already have Identity Authentication (a mandatory service), which enables you to sign in and authenticate in the Identity Provisioning UI and within SAP Business Technology Platform.

To obtain Identity Provisioning, you need to create an incident. Follow the steps:

1. Create an incident to component [BC-IAM-IPS](#).
2. Explain that you've purchased an SAP Jam Collaboration product and you require access to the Identity Provisioning service.
3. Specify the S-user to be assigned as the first administrator of the Identity Provisioning tenants. Later, this S-user can add other users as administrators.

## **How To Use**

If your SAP Jam Collaboration license includes Identity Provisioning, this bundle tenant is provisioned to your organization with preconfigured source and target systems. Identity Authentication system is preconfigured as a source system and SAP Jam Collaboration is preconfigured as a target. For more information, see [Identity Authentication and Identity Provisioning](#) in the *SAP Jam Collaboration Administrator Guide*.

You can review the provisioning system configurations, adjust them if needed and schedule read jobs.

## **Related Information**

[SAP Jam Collaboration](#)

## 1.5.6 SAP Marketing Cloud Bundle

SAP Marketing Cloud bundle allows you to use the Identity Provisioning service for synchronizing user data between source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

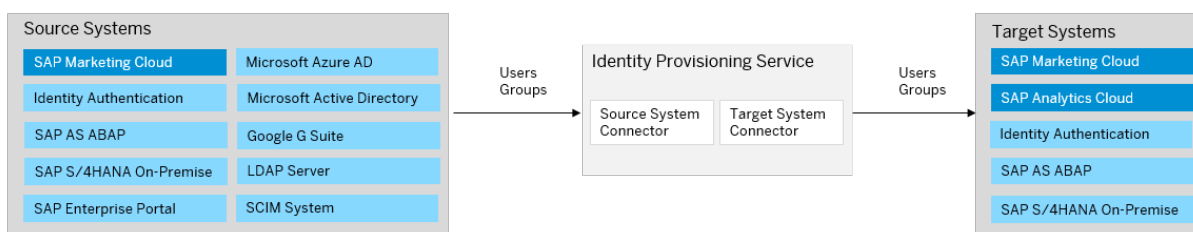
### Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

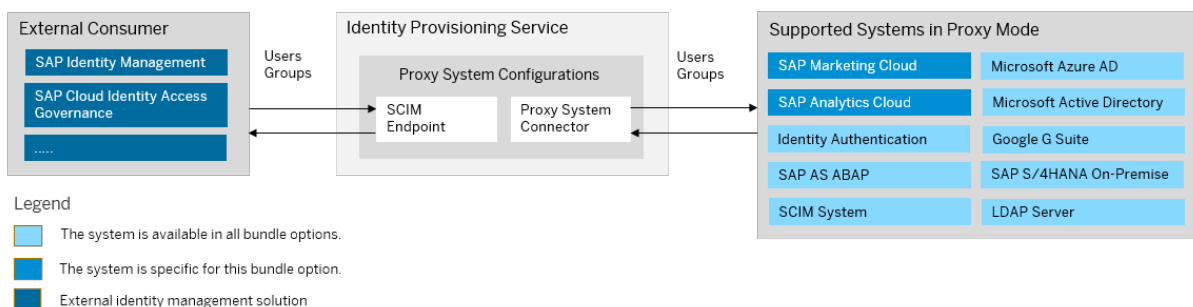
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

### Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



## You have purchased the product before release 1911

If you've bought SAP Marketing Cloud before the **1911** release, you have to request access to the Identity Provisioning service. To do this, follow the steps:

1. Create an incident to component *XX-S4C-OPR-SRV (S/4HANA Cloud service requests)*.
2. Explain that you've purchased an SAP Marketing Cloud product before **1911** release.
3. Specify the S-user to be assigned as the first administrator of the Identity Provisioning tenants. Later, this S-user can add other users as administrators.

4. Specify the URLs to your SAP Marketing Cloud **Quality** and **Productive** systems.

In the reply of your incident, you'll receive two URLs related to two Identity Provisioning tenants. The first URL will be bound to your **Quality** instance, and you can use it for [testing](#) purposes. The second URL will be bound to your **Productive** instance, and you can use it for [productive](#) provisioning configurations and jobs. This bounding principle is applied to your Identity Authentication tenants as well.

## You have purchased the product after release 1911

If you've bought SAP Marketing Cloud with release **1911** or higher, you'll receive two onboarding e-mails from SAP. According to your contract with SAP, a technical contact person has been chosen as the first user of the Identity Provisioning service, who is granted with [Administrator](#) permissions. Each onboarding e-mail contains a URL link that you, as an administrator, can use to directly access the Identity Provisioning UI. The relevant URLs are related to two different Identity Provisioning tenants – the first one you can use for [testing](#) purposes, and the second one – for [productive](#) provisioning configurations and jobs.

If you encounter issues with accessing your Identity Provisioning UI, create an incident to component [BC-IAM-IPS](#).

## Related Information

[SAP Marketing Cloud](#)

## 1.5.7 SAP SuccessFactors Bundle

SAP SuccessFactors bundles with SAP Cloud Identity Services – Identity Authentication and Identity Provisioning.

If your SAP SuccessFactors tenant was created after December 9, 2022, Identity Authentication and Identity Provisioning have already been enabled. You do not need to complete the steps to upgrade to Identity Authentication, as described in [Initiating the Upgrade to SAP Cloud Identity Services - Identity Authentication Service](#).

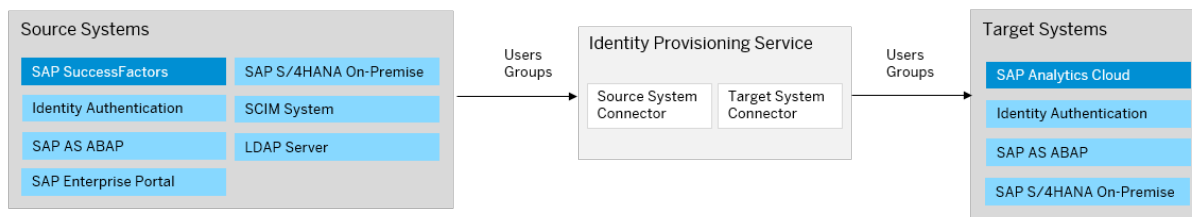
### i Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

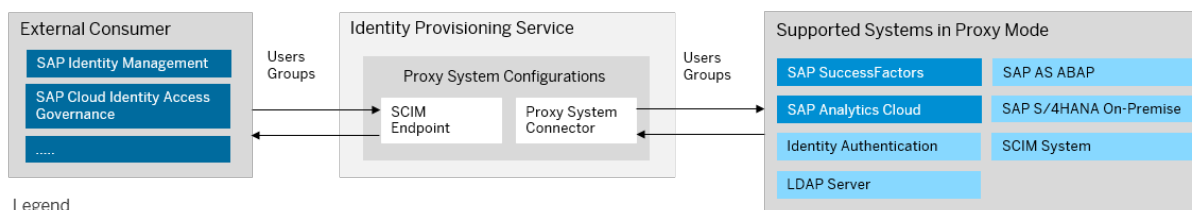
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

## Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



### Legend

- The system is available in all bundle options.
- The system is specific for this bundle option.
- External identity management solution

### Note

SAP SuccessFactors target system is enabled only for Identity Provisioning bundle tenants running on SAP Cloud Identity Services infrastructure.

## How to Obtain

After purchasing SAP SuccessFactors, if your license includes only SAP SuccessFactors, you can obtain Identity Authentication and Identity Provisioning as well. To do this, you have to enable these services via the [Upgrade Center](#) of SAP SuccessFactors. You need to provide an S-user and password for validation. This S-user will be assigned as the first administrator of the Identity Provisioning tenants.

### Note

In case you already have an existing Identity Provisioning tenant, its initial administrator may be different than the S-user you've provided in the [Upgrade Center](#) procedure. Due to security reasons, this S-user will not be assigned as an administrator. If you want it to be granted IPS\_ADMIN permissions, you need to ask the initial Identity Provisioning administrator to do that for you. To learn how, see: [Manage Authorizations \[page 1487\]](#)

See also: [\(Guided Answers\) Can't Get Admin Permissions for a Bundle](#)

You also need to create a password for the automatically created SAP SuccessFactors API user (*IPSADMIN*), and grant it the following permissions:

- Go to **Admin Center** > **Manage Permission Roles** > **Manage Integration Tools** and choose "Allow Admin to Access OData API through Basic Authentication".

- Go to ► [Admin Center](#) ► [Manage Permission Roles](#) ► [Administrator Permission](#) ► [Manage User](#) ► and choose "User tenant OData entity".

Proceed as follows:

1. [Getting Started with Identity Authentication and SAP SuccessFactors](#)
2. [Initiate Upgrade to Identity Authentication](#)
3. [Identity Provisioning Administration Console Tasks](#)
4. [Setting an API User for Sync Jobs](#)

Your Identity Provisioning system is created once you've initiated your upgrade to Identity Authentication, and the Identity Provisioning administrator receives an email with two links to the Identity Provisioning admin console (related to two different tenants) – one for [testing](#) purposes, and one – for [productive](#) provisioning configurations and jobs.

If the upgrade procedure fails, create an incident to component [LOD-SF-PLT-IAS](#).

If you have troubles with the Identity Provisioning admin console or configurations, create an incident to component [BC-IAM-IPS](#).

## How to Use

This bundle tenant is provisioned to your organization with preconfigured source and target systems. SAP SuccessFactors is preconfigured as a source system and Identity Authentication is preconfigured as a target.

Integrating SAP SuccessFactors with the embedded SAP Analytics Cloud is triggered from SAP SuccessFactors [Upgrade Center](#). As a result, SAP Analytics Cloud is created as a target system. It is disabled and no source system is linked to it.

You can review the provisioning system configurations, adjust them if needed and schedule read jobs.

## Related Information

[SAP SuccessFactors Platform](#)

## 1.5.8 SAP SuccessFactors Learning Bundle

SAP SuccessFactors Learning bundle allows you to use the Identity Provisioning service for synchronizing user data between particular source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

### Note

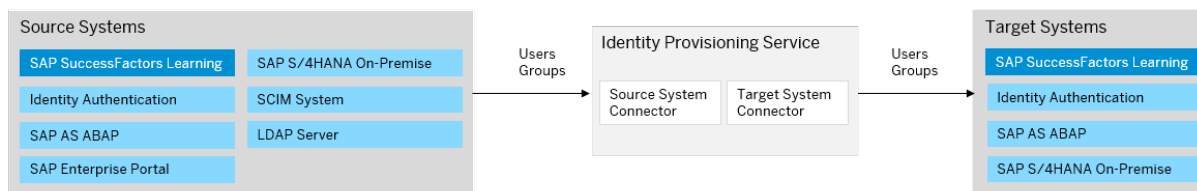
As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).



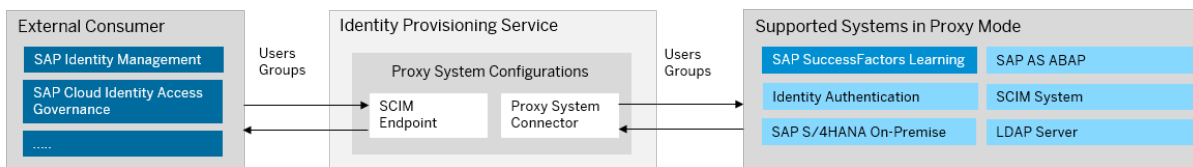
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

## Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



Legend

- The system is available in all bundle options.
- The system is specific for this bundle option.
- External identity management solution

## How to Obtain

After purchasing SAP SuccessFactors with license for SAP SuccessFactors Learning, you can obtain Identity Authentication and Identity Provisioning tenants by initiating the integration in SAP SuccessFactors [Upgrade Center](#) as outlined here: [How to Integrate SAP SuccessFactors Learning and SAP Cloud Identity Services](#)

The process to integrate SAP SuccessFactors Learning and SAP Cloud Identity Services - Identity Authentication and Identity Provisioning consists of the following four steps:

1. [Initiating Learning Identity Authentication Migration](#)
2. [Learning Configuration Procedure](#)
3. [Migrating Learning Users to Identity Authentication](#)
4. [Identity Authentication Activation](#)

### Note

If you have troubles with the Identity Provisioning admin console or configurations, create an incident to component [BC-IAM-IPS](#).

## How to Use

This bundle tenant is provisioned to your organization with preconfigured source and target systems. SAP SuccessFactors Learning is preconfigured as a source system and Identity Authentication is preconfigured as a target.

You can review the provisioning system configurations, adjust them if needed and schedule read jobs.

### 1.5.9 SAP S/4HANA Cloud Bundle

SAP S/4HANA Cloud bundle allows you to use the Identity Provisioning service for synchronizing user data between source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

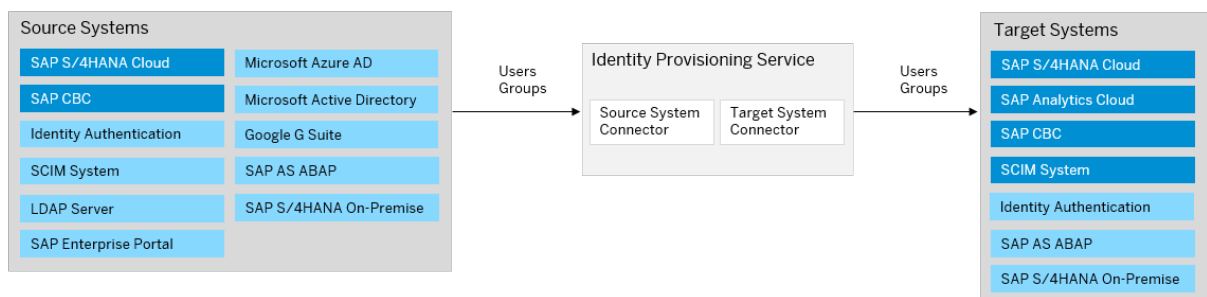
#### Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

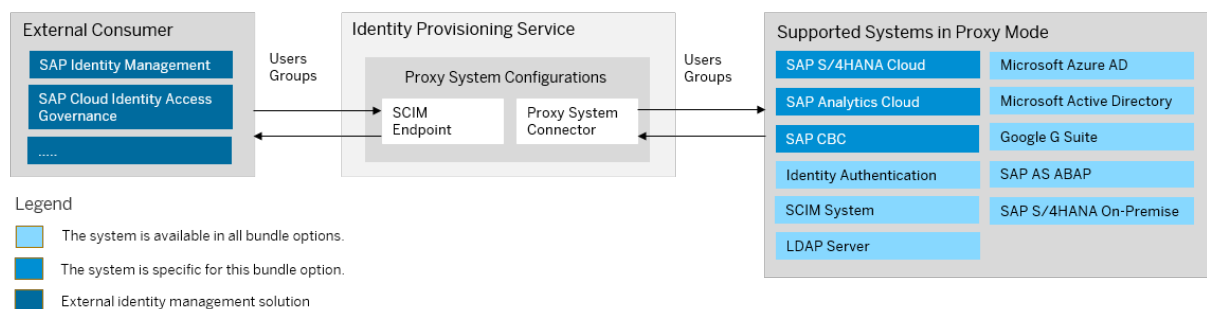
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

#### Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



### i Note

This bundle option is only applicable for [SAP S/4HANA Cloud Essentials Edition](#) .

### ! Restriction

**SCIM System** can be used as a target system only in provisioning scenarios with [SAP Central Business Configuration](#).

## How to Obtain

The way you obtain Identity Provisioning bundle tenant depends on the release of your SAP S/4HANA Cloud.

### SAP S/4HANA Cloud Purchased Before 1911 Release

If you've purchased SAP S/4HANA Cloud before the **1911** release, you have to request access to the Identity Provisioning service. To do this, follow the steps:

1. Create an incident to component [XX-S4C-OPR-SRV \(S/4HANA Cloud service requests\)](#).
2. Explain that you've purchased an SAP S/4HANA Cloud product before **1911** release.
3. Specify the S-user to be assigned as the first administrator of the Identity Provisioning tenants. Later, this S-user can add other users as administrators.

In the reply of your incident, you'll receive two URLs related to two Identity Provisioning tenants. The first URL will be bound to your **Quality** instance, and you can use it for [testing](#) purposes. The second URL will be bound to your **Productive** instance, and you can use it for [productive](#) provisioning configurations and jobs. This bounding principle is applied to your Identity Authentication tenants as well.

### SAP S/4HANA Cloud Purchased With 1911 Release or Higher

If you've purchased SAP S/4HANA Cloud with release **1911** or higher, you'll receive two onboarding e-mails from SAP. According to your contract with SAP, a technical contact person has been chosen as the first user of the Identity Provisioning service, who is granted with [Administrator](#) permissions. Each onboarding e-mail contains a URL link that you, as an administrator, can use to directly access the Identity Provisioning UI. The relevant URLs are related to two different Identity Provisioning tenants – the first one you can use for [testing](#) purposes, and the second one – for [productive](#) provisioning configurations and jobs.

## How to Use

The Identity Provisioning bundle tenant for SAP S/4HANA Cloud release 1911 or higher comes with preconfigured provisioning systems (connectors) for specific integration scenarios:

- For integrating SAP S/4HANA Cloud with the embedded SAP Analytics Cloud, SAP provides preconfigured SAP S/4HANA Cloud source system and SAP Analytics Cloud target system. To learn more about what's been initially preconfigured and how to apply subsequent changes to the Identity Provisioning configurations, see:
  - [Ensure SAP S/4HANA Cloud users are replicated during their validity period](#)

- [Ensure deleted SAP S/4HANA Cloud users are also deleted in SAP Analytics Cloud](#)
- For integrating SAP S/4HANA Cloud with Identity Authentication, SAP provides preconfigured Identity Authentication source system and SAP S/4HANA Cloud target system. Also preconfigured Identity Authentication source system and SAP Central Business Configuration target system are delivered with system provisioning.

#### i Note

Your initial provisioning jobs are paused. This means, you need to manually run them. To do this, open your relevant source systems, go to the ► [Jobs](#) ► [Read Job](#) ► [Schedule](#) ► and turn the job scheduler **ON**. For more information, see [Read Provisioning Job \[page 1526\]](#). If you need to, correct the credentials (users and passwords) for your relevant systems first, and then start the initial job(s).

If you encounter issues with accessing your Identity Provisioning UI, create an incident to component [BC-IAM-IPS](#).

## Related Information

[SAP S/4HANA Cloud](#)

[SAP Identity Provisioning \(IPS\) is now bundled with SAP S/4HANA Cloud!](#) 

## 1.5.10 SAP Build Work Zone, advanced edition Bundle

SAP Build Work Zone, advanced edition bundle allows you to use the Identity Provisioning service for synchronizing user data between source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

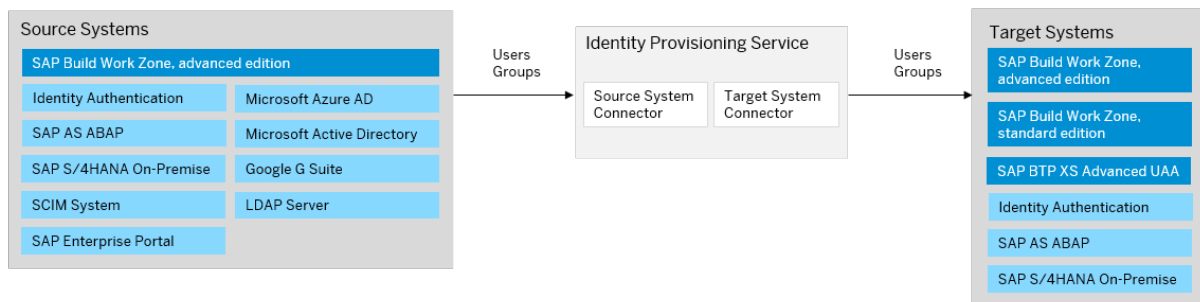
#### i Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

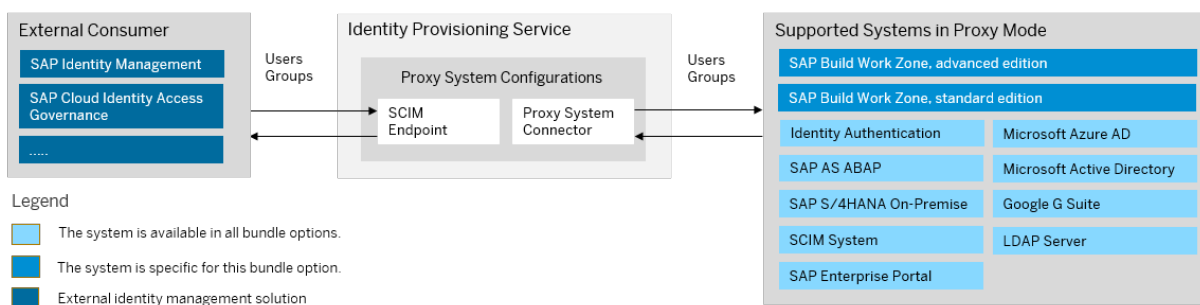
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

## Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



## How to Obtain

After purchasing SAP Build Work Zone, advanced edition, you can obtain the Identity Provisioning tenant by connecting your subaccount to the service in Work Zone Manager. As the subaccount administrator, proceed as follows:

1. Connect your subaccount to Identity Provisioning as described in: [Post Booster Configuration](#)
2. Open the URL <https://iamtenants.accounts.cloud.sap/> and login with your S-user.
3. View the Identity Provisioning and Identity Authentication tenants that are assigned to your customer ID.

For the complete onboarding process, see [Getting Started](#)

## Related Information

[SAP Build Work Zone, advanced edition](#)

## 1.5.11 Sales Cloud – Analytics & AI Bundle

Sales Cloud – Analytics & AI bundle allows you to use the Identity Provisioning service for synchronizing user data between source and target systems. The available source and target systems in this bundle can also be configured as proxy systems for indirect connection to external identity management systems.

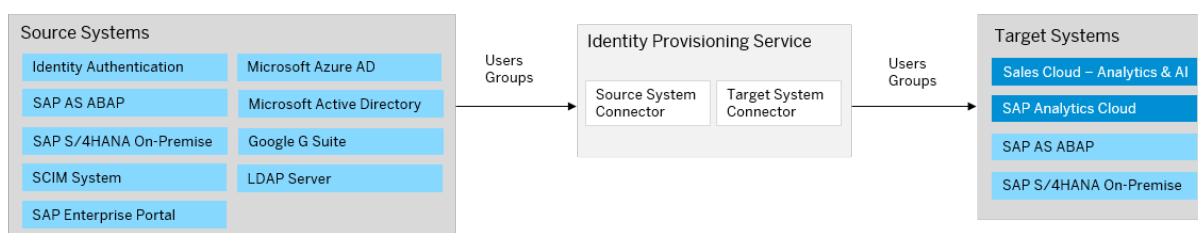
### Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

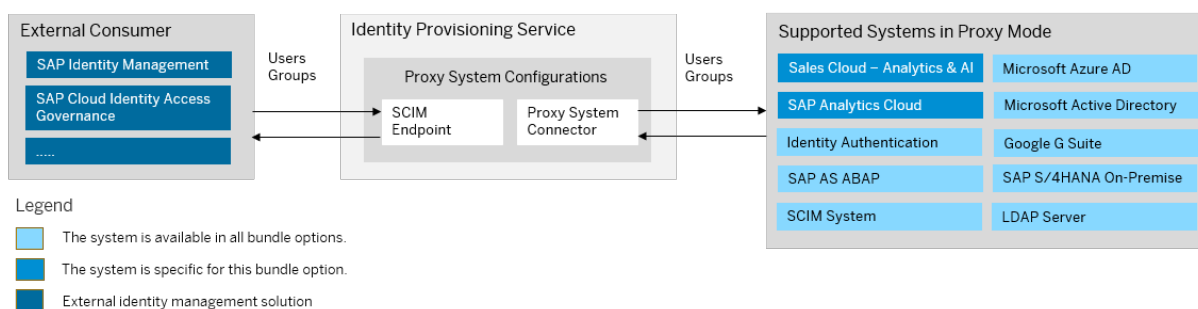
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

### Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



## How to Obtain

After purchasing Sales Cloud – Analytics & AI, you'll receive two e-mails from SAP. According to your contract with SAP, a technical contact person has been chosen as the first user of the Identity Provisioning service, who is granted with [Administrator](#) permissions. In these e-mails from SAP, you'll find the ID of this administrator (their P- or S-user) and their e-mail address. They can access the Identity Provisioning UI with their user credentials.

Each e-mail from SAP contains also a URL link that you, as an administrator, can use to directly access the Identity Provisioning UI. These two URLs are related to two different Identity Provisioning tenants – the first one you can use for [testing](#) purposes, and the second one – for [productive](#) provisioning configurations and jobs.

## 1.5.12 SAP Concur Bundle

SAP Concur bundles with SAP Cloud Identity Services – Identity Authentication and Identity Provisioning.

After purchasing the cloud-based travel and expense management solution, you are entitled to receive Identity Authentication and Identity Provisioning tenants. During the SAP Concur order fulfilment process, it is checked whether you, as a customer, already have SAP Cloud Identity Services tenants. If not, those are created without additional costs on the purchase of SAP Concur license.

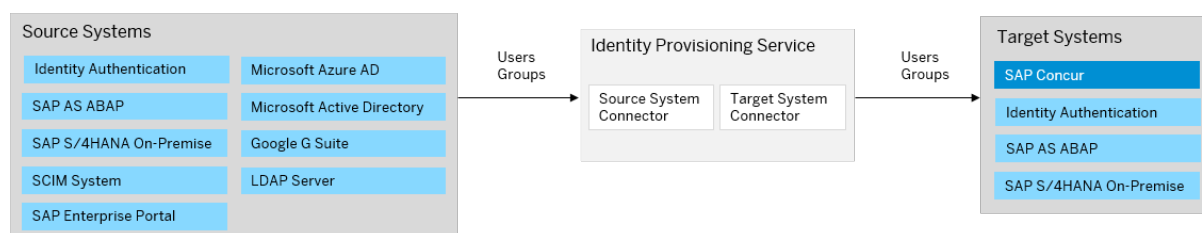
### Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

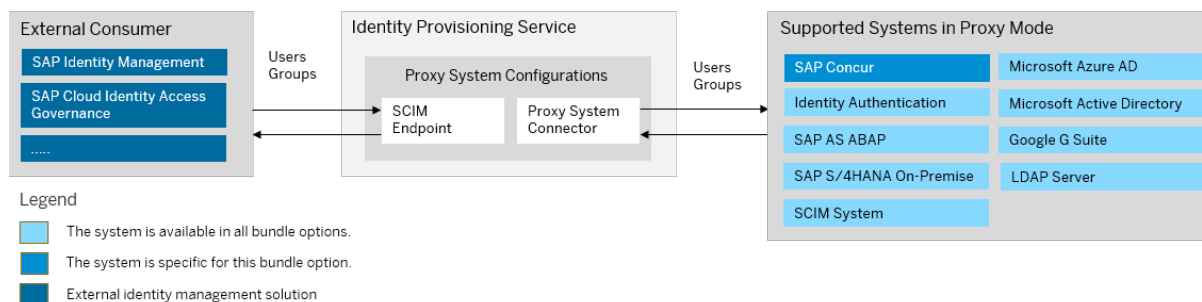
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

### Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



## How to Obtain

You'll receive two onboarding e-mails from SAP. According to your contract with SAP, a technical contact person has been chosen as the first user of the Identity Provisioning service, who is granted with *Administrator* permissions. Each onboarding e-mail contains a URL link that the administrator can use to directly access the Identity Provisioning UI. The relevant URLs are related to two different Identity Provisioning tenants - the first one you can use for *testing* purposes, and the second one - for *productive* provisioning configurations and jobs.

## How to Use

Normally, the Identity Authentication and Identity Provisioning bundle tenants are pre-configured for out-of-the-box integrations between SAP solutions. However, in the case of SAP Concur bundle, you need to **manually configure** both tenants for your authentication and provisioning scenarios. For more information, see [SAP Concur Integration Scenario](#)

### 1.5.13 SAP Fieldglass Bundle

SAP Fieldglass bundles with SAP Cloud Identity Services – Identity Authentication and Identity Provisioning.

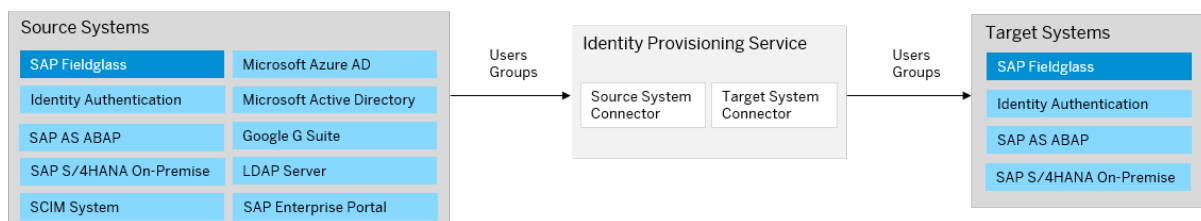
#### Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

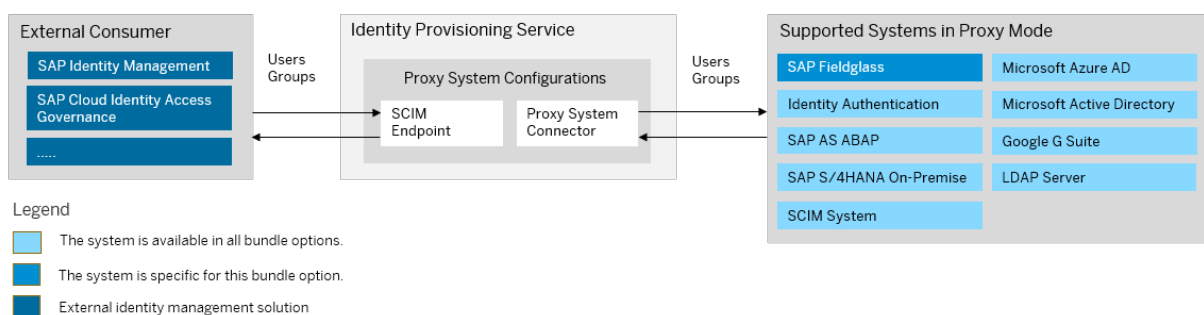
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

#### Bundle Tenant on Neo Environment

Default Mode: Provision user data from source to target systems



Proxy Mode: Provision user data to and from central identity management solution and systems with proxy configuration



## How to Obtain

After purchasing SAP Fieldglass, the technical contact person of your organization receives two onboarding e-mails from SAP. Each of them provides a tenant URL for accessing the SAP Cloud Identity Services



administration console. One of the tenant URLs is for testing purposes, the other one is for productive usage. The technical contact person is granted the administrator permissions of the tenants and performs the initial logon to the SAP Cloud Identity Services administration console.

## How to Use

This bundle tenant is provisioned to your organization with preconfigured source and target systems.

- If you are a new SAP Fieldglass customer (greenfield scenario), you get Identity Authentication preconfigured as a source system and SAP Fieldglass preconfigured as a target.
- If you are an existing SAP Fieldglass customer (brownfield scenario), in addition to preconfigured Identity Authentication source system and SAP Fieldglass target system, you get preconfigured SAP Fieldglass source system and Identity Authentication target system. To use brownfield scenario, your SAP Fieldglass administrator must trigger an upgrade process from the SAP Fieldglass UI.

You can review the provisioning system configurations, adjust them if needed and schedule read jobs.

## Related Information

[SAP Fieldglass and SAP Cloud Identity Services Business Synopsis](#)

### 1.5.14 SAP Business Ecology Management Bundle

SAP Business Ecology Management (also known as SAP Sustainability Footprint Management) bundles with SAP Cloud Identity Services - Identity Authentication and Identity Provisioning.

#### i Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

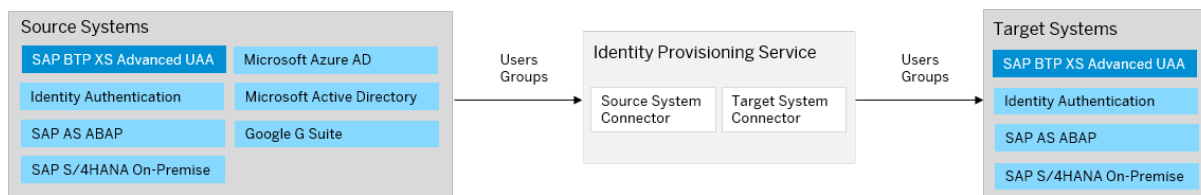
## How to Obtain

After purchasing the SAP cloud solution that helps customers assess and analyze their environmental impact, the technical contact person of your organization receives two onboarding e-mails from SAP. Each of them provides a tenant URL for accessing the SAP Cloud Identity Services administration console. One of the tenant URLs is for testing purposes, the other one is for productive usage. The technical contact person is granted

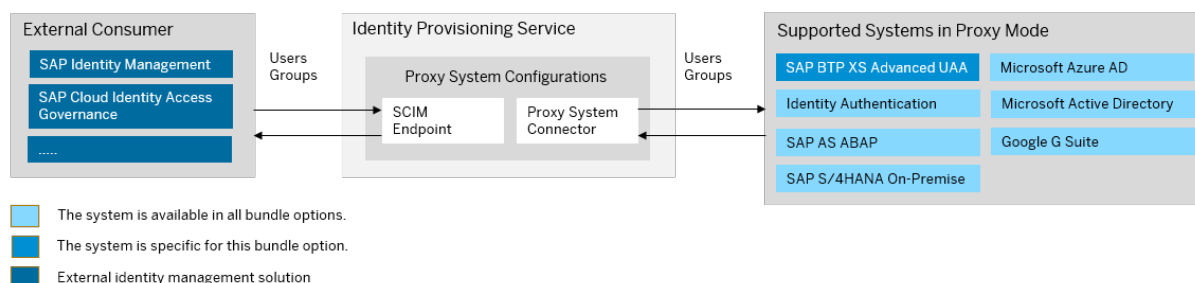
the administrator permissions of the tenants and performs the initial login to the SAP Cloud Identity Services administration console.

## Bundle Tenant on Neo Environment

**Default Mode:** Provision user data from source to target systems



**Proxy Mode:** Provision user data to and from central identity management solution and systems with proxy configuration



## How to Use

This bundle tenant is provisioned to your organization with Identity Authentication default user groups created for SAP Business Ecology Management. Your source and target systems are not preconfigured. You need to add and configure them manually.

## 1.5.15 SAP Commerce Cloud Bundle

SAP Commerce Cloud bundles with SAP Cloud Identity Services – Identity Authentication and Identity Provisioning.

### i Note

As of March 15, 2022, Identity Provisioning bundle tenants are created on the infrastructure of SAP Cloud Identity Services only. These tenants have most of the provisioning systems (connectors) enabled by default, with a few exceptions described in [Connectors Availability in Bundle Tenants on SAP Cloud Identity Infrastructure](#).

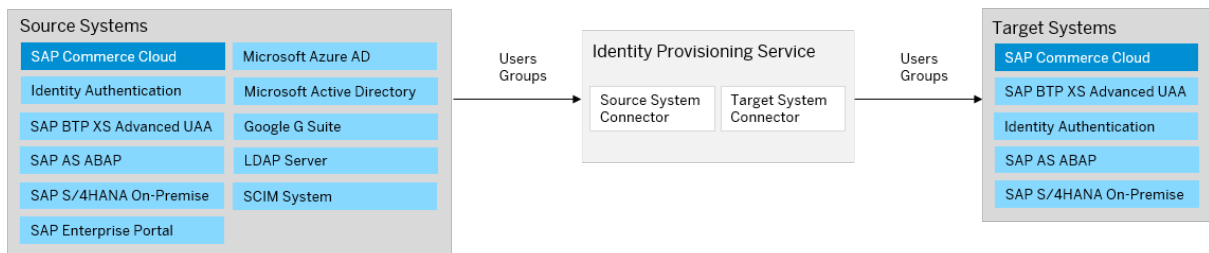
Identity Provisioning bundle tenants running on SAP BTP, Neo environment have a limited number of connectors enabled by default. They are illustrated in the diagram below.

## How to Obtain

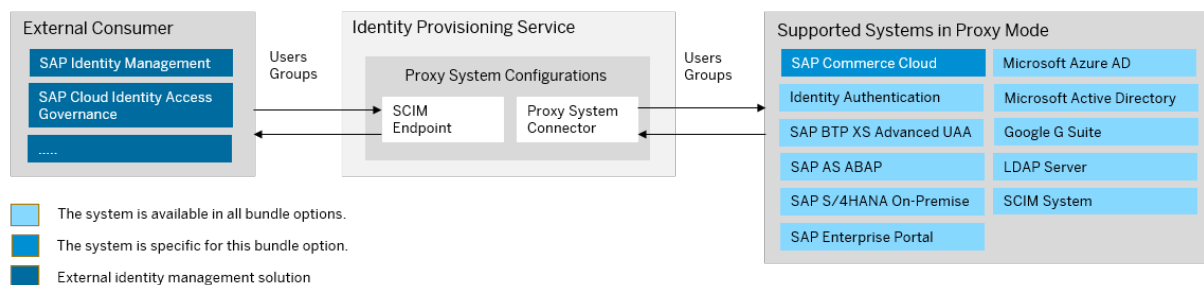
After purchasing SAP Commerce Cloud, the technical contact person of your organization receives two onboarding e-mails from SAP. Each of them provides a tenant URL for accessing the SAP Cloud Identity Services administration console. One of the tenant URLs is for testing purposes, the other one is for productive usage. The technical contact person is granted the administrator permissions of the tenants and performs the initial logon to the SAP Cloud Identity Services administration console.

## Bundle Tenant on Neo Environment

**Default Mode:** Provision user data from source to target systems



**Proxy Mode:** Provision user data to and from central identity management solution and systems with proxy configuration



## How to Use

This bundle tenant is provisioned to your organization with preconfigured source and target systems. Identity Authentication system is preconfigured as a source and SAP Commerce Cloud solution is preconfigured as a target. Identity Authentication default user groups are created for SAP Commerce Cloud.

You can review the provisioning system configurations, adjust them if needed and schedule read jobs.

## 1.6 Supported Systems

The Identity Provisioning service supports provisioning of users and groups between multiple supported cloud and on-premise systems, both SAP and non-SAP.

For more information about all provisioning systems (connectors) which you can use as *source*, *target*, and *proxy* system types for your provisioning scenarios, refer to the topics in this section.

For more information about the availability of the provisioning systems in bundle tenants, see [Bundle Tenants and Connectors](#) [page 422].

For more information about the availability of the provisioning systems in standalone tenants, see [Standalone Tenants](#) [page 1608].

### Related Information

[Source Systems](#) [page 452]

[Target Systems](#) [page 702]

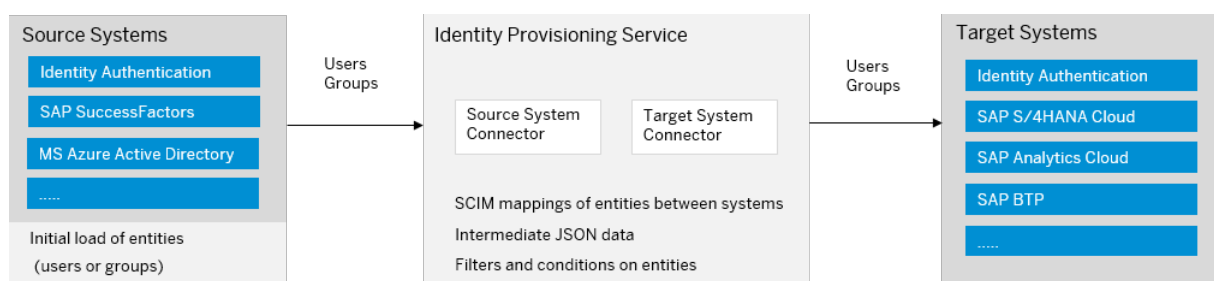
[Proxy Systems](#) [page 981]

### 1.6.1 Source Systems

A source system is the connector used for reading entities (users, groups, roles).

Source systems can be on-premise or cloud-based, SAP or non-SAP, and usually represent the corporate user store where identities are currently maintained. The Identity Provisioning service reads the entities from the source system and creates or updates them in the relevant target ones. The provisioning is triggered from the *Jobs* tab of a source system.

You can connect one source system to one or multiple target systems. In the case of multiple (enabled) target systems, when you start a *Read* or a *Resync* job, this operation will trigger provisioning of entities from this source system to all relevant target ones.



#### ! Restriction

By default, the maximum number of productive source systems you are allowed to add for your tenant is 20.

If your business requires using more systems, create an incident for component [BC-IAM-IPS](#) to request them. Describe your scenarios and provide a reason why you need the additional systems.

## Related Information

[System Types \[page 86\]](#)

### 1.6.1.1 Identity Authentication

Follow this procedure to set up SAP Cloud Identity Service – Identity Authentication as a source system.

## Prerequisites

To establish the connection between Identity Provisioning and Identity Authentication, you need to set up the technical user (of type [System](#)) in Identity Authentication and assign this user the necessary authorizations. You can do it now (as a prerequisite) or in the process of configuring Identity Authentication as a source system, as described in step 3.

## Context

Identity Authentication provides authentication and single sign-on for users in the cloud. The user store of Identity Authentication can manage different type of users (employees, partners, customers and public) as well as groups. Using Identity Provisioning, you can read those users (self-registered, imported, or manually created) and groups and provision them to various target systems.

For this, you need to configure the Identity Authentication as a source system in the Identity Provisioning UI. This source system consumes SCIM 2.0 API provided by Identity Authentication.

There are two versions of the Identity Authentication SCIM API. They are handled by the `ias.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on July 9, 2021) - Identity Authentication SCIM API (in short, SCIM API version 1) is used. For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#)
- When the value is set to **2** - Identity Directory SCIM API (in short, SCIM API version 2) is used. This value is set automatically for all manually created systems in the Identity Provisioning UI after versioning was introduced on July 9, 2021.  
SCIM API version 2 is enhanced to support paging for group members and user's groups, custom attributes, delta read mode for users. Also, the group resource mapping in the transformation is not ignored by default, as it is in SCIM API version 1.

### i Note

Identity Authentication (using SCIM API version 2) and Identity Directory are sometimes used interchangeably. Identity Directory is the persistency layer of SAP Cloud Identity Services – Identity Authentication.




To create Identity Authentication as a source system, proceed as follows:

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *Identity Authentication* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and Identity Authentication and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added Identity Authentication source system, select the *Certificate* tab and choose  *Generate*  *Download* , as described in [Manage Certificates \[page 1506\]](#).

Skip step **a.** if you want to use basic authentication.

In SAP Cloud Identity Services administration console, perform the next steps. They are relevant for both basic and certificate-based authentication.

- b. [Add System as administrator](#) and provide the respective credentials.

For basic authentication, provide a password. The user ID will be generated automatically when you set the password for the first time.

For certificate-based authentication, upload the certificate you have generated in SAP Cloud Identity Services administration console on the previous step.

- c. Save your changes.
  - d. Make sure [Manage Users](#) (or at least [Read Users](#)) and [Manage Groups](#) authorization roles are enabled for the technical user. This way, you can read users and groups from the Identity Authentication user store.
4. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL of the Identity Authentication tenant of your company.</p> <p>For example: <b>https://mytenant.accounts.on-demand.com</b></p> <div> <p><b>Note</b></p> <p>If your Identity Authentication Shanghai (China) tenants reside on SAP BTP, Neo environment, you should use the following URL pattern: <a href="https://&lt;tenant_ID&gt;.accounts.sapcloud.cn/">https://&lt;tenant_ID&gt;.accounts.sapcloud.cn/</a></p> </div>
ProxyType	<p>Enter: <a href="#">Internet</a></p> <p>The Identity Authentication service is a cloud solution and is outside of your company on-premise infrastructure.</p>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"> <li>• <a href="#">BasicAuthentication</a></li> <li>• <a href="#">ClientCertificateAuthentication</a></li> </ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the Client ID (previously User ID) of the Identity Authentication technical user. It is generated automatically for the administrator of type system, when choosing <a href="#">Secrets</a> <a href="#">Add</a> <a href="#">Save</a>. For example: <a href="#">1ab7c243-5de5-4530-8g14-1234h26373ab</a></p> <p>If your technical user was created before <a href="#">January 2020</a>, enter the T-user. For example: <a href="#">T000003</a></p>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the Client Secret (previously Password) of the Identity Authentication technical user. It is generated automatically for the administrator of type system, when choosing <a href="#">Secrets</a> <a href="#">Add</a> <a href="#">Save</a>.</p>

## Optional Properties

Property Name	Description & Value
<ul style="list-style-type: none"> <li>ias.&lt;property_name&gt;</li> <li>scim.&lt;property_name&gt;</li> </ul>	<p>When using SCIM API version 2, property names start with <code>ias</code> prefix, for example: <code>ias.user.unique.attribute</code>.</p> <p>When using SCIM API version 1, property names start with <code>scim</code> prefix, for example: <code>scim.user.unique.attribute</code>.</p> <p>For more information, see <a href="#">List of Properties [page 94]</a>. Use the main search or filter properties by <i>Name</i> or <i>System Type</i> columns.</p>
<code>ias.user.filter</code>	<p>When specified, only those users matching the filter expression will be read.</p> <p>For example: <b>name.familyName eq "Smith" and addresses.country eq "US"</b></p> <p>This filter will read only users whose name is "<i>Smith</i>" and are living in the <i>United States</i>.</p> <p>For more information, see <a href="#">Identity Directory SCIM API: User Search</a>.</p>
<code>ias.group.filter</code>	<p>When specified, only those groups matching the filter expression will be read.</p> <p>For example: <b>displayName eq "ProjectTeam1"</b></p> <p>This filter will read only groups, whose display name is "<i>ProjectTeam1</i>".</p> <p>For more information, see <a href="#">Identity Directory SCIM API: Group Search</a>.</p>
<code>ips.failed.request.retry.attempts</code>	Predefined value: <code>2</code>
<code>ips.failed.request.retry.attempts.interval</code>	Predefined value: <code>60</code>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Identity Authentication* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

When Identity Authentication is configured as a source system, the default transformation logic reads all the user attributes from the Identity Authentication user store. The logic is provided by the Identity Authentication SCIM REST API, which then maps the attributes to the internal SCIM representation.



You can change the default transformation mapping rules to reflect your current setup of entities in your Identity Authentication system. For more information, see: [Manage Transformations \[page 1494\]](#)

SCIM API version 1: [Identity Authentication: SCIM REST API](#)

SCIM API version 2: [Identity Directory SCIM API](#) 

### Note

When a user is deleted from the Identity Authentication, the deletion status is considered by it during the read processes. Depending on the off-boarding user handling in the target system, a user can be deleted, or can be set to *inactive*.

### Default transformation for SCIM API version 1:

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userUuid",
        "targetPath": "$[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.name.honorificPrefix",
        "targetPath": "$.name.honorificPrefix",
        "optional": true
      },
      {
        "sourcePath": "$.emails[*].value",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails[?(@.value)]"
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath": "$.emails[?(@.primary== true)].value",
      "correlationAttribute": true
    },
    {
      "sourcePath": "$.active",
      "targetPath": "$.active",
      "optional": true,
      "defaultValue": true
    },
    {
      "sourcePath": "$.userType",
      "targetPath": "$.userType",
      "optional": true
    },
    {
      "sourcePath": "$.addresses",
      "targetPath": "$.addresses",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "optional": true
    },
    {
      "sourcePath": "$.phoneNumbers",
      "targetPath": "$.phoneNumbers",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "sourcePath": "$.timeZone",
      "targetPath": "$.timezone",
      "optional": true
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "optional": true
    },
    {
      "sourcePath": "$.sourceSystemId",
      "targetPath": "$.sourceSystemId",
      "ignore": true
    },
    {
      "sourcePath": "$.groups",
      "targetPath": "$.groups",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "type": "remove",
      "targetPath": "$.groups[*].display"
    },
    // The display name of the corporate groups is removed because it's not
    // necessary for the target system API.
    {
      "condition": "$.displayName EMPTY true",
      "type": "remove",
      "targetPath": "$.displayName"
    },
  ],

```

```

        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['value']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['displayName']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['displayName']",
        "optional" : true
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']",
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']",
        "optional" : true
    }

```

```

        },
        {
            "sourcePath": "$.company",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
            "optional": true
        }
    ]
},
// By default, the group mapping is inactive (ignored) but groups are
supported.
// To start reading groups, either delete the statement "ignore: true", or
set its value to false.
"group": {
    "ignore": true,
    "mappings": [
        {
            "sourcePath": "$.id",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath": "$.schemas[0]"
        },
        {
            "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
            "targetPath": "$.displayName"
        },
        {
            "optional": true,
            "preserveArrayWithSingleElement": true,
            "sourcePath": "$.members",
            "targetPath": "$.members"
        },
        {
            "ignore": true,
            "constant":
"urn:sap:cloud:scim:schemas:extension:custom:2.0:Group",
            "targetPath": "$.schemas[1]"
        },
        {
            "ignore": true,
            "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
            "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']"
        },
        {
            "ignore": true,
            "optional": true,
            "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['description']",
            "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['description']"
        }
    ]
}
}

```

Default transformation for SCIM API version 2:

## Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem",
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['userId']"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userId']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userId']"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['uuid']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['uuid']"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.name.honorificPrefix",
        "targetPath": "$.name.honorificPrefix",
        "optional": true
      },
      {
        "sourcePath": "$.emails[*].value",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails[?(@.value)]"
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      }
    ]
  }
}
```

```

    {
      "sourcePath": "$.userType",
      "targetPath": "$.userType",
      "optional": true
    },
    {
      "sourcePath": "$.addresses",
      "targetPath": "$.addresses",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "optional": true
    },
    {
      "sourcePath": "$.phoneNumbers",
      "targetPath": "$.phoneNumbers",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "sourcePath": "$.timezone",
      "targetPath": "$.timezone",
      "optional": true
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "optional": true
    },
    {
      "sourcePath": "$.groups",
      "targetPath": "$.groups",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "type": "remove",
      "targetPath": "$.groups[*].display"
    },
    {
      "condition": "$.displayName EMPTY true",
      "type": "remove",
      "targetPath": "$.displayName"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystem']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystem']",
      "optional": true
    },
  },

```

```

        {
            "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId'],
            "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId'],
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",

```

```

        "optional": true
      },
      {
        "sourcePath": "$",
        "targetPath": "$",
        "optional": true
      },
      {
        "sourcePath": "$.company",
        "targetPath": "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']",
        "optional": true
      }
    ],
    "group": {
      "mappings": [
        {
          "sourcePath": "$.id",
          "targetVariable": "entityIdSourceSystem"
        },
        {
          "sourcePath": "$",
          "targetPath": "$",
          "optional": true
        },
        {
          "sourcePath": "$.displayName",
          "targetPath": "$.displayName",
          "optional": true
        },
        {
          "sourcePath": "$.members",
          "targetPath": "$.members",
          "preserveArrayWithSingleElement": true,
          "optional": true
        }
      ]
    }
  }
}

```

6. Add a target system where to provision users and groups. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Identity Authentication: Documentation](#)



## 1.6.1.2 Local Identity Directory

Follow this procedure to set up Local Identity Directory as a source system.

### Prerequisites

#### Note

The *Local Identity Directory* connector is available for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

### Context

Identity directory is the user store of SAP Cloud Identity Services. It provides a central place for storing and managing users, groups and custom schemas through the System for Cross-domain Identity Management 2.0 REST API, in short Identity Directory SCIM API.

You can read users and groups from the identity directory and provision them to various target systems by configuring the Local Identity Directory connector as a source system in the SAP Cloud Identity Services administration console. The entities are read from the identity directory of your current SAP Cloud Identity Services tenant. Therefore, you don't have to set up connectivity and authentication properties for the source system. In case you want to read entities from the identity directory in another tenant, add Identity Authentication (version 2) as a source system and configure the respective connectivity and authentication properties.

To create Local Identity Directory as a source system, proceed as follows:

### Procedure


1. Sign in to SAP Cloud Identity Services administration console and navigate to ► [Identity Provisioning](#) ► [Source Systems](#) .
2. Add *Local Identity Directory* as a source system.  
For more information, see [Add a System \[page 1477\]](#).
3. **Optional:** Configure properties. For example, filtering properties to control the entities to be provisioned:  
`ids.user.filter`.  
Local Identity Directory properties are prefixed with `ids.<property_name>`. For more information, see [List of Properties \[page 94\]](#).

#### 4. **Optional:** Configure transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the Local Identity Directory source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

When Local Identity Directory is configured as a source system, the default transformation logic reads the user attributes from the user store of SAP Cloud Identity Services. The logic is provided by the Identity Directory SCIM API, which then maps the attributes to the internal SCIM representation.

You can change the default transformation mapping rules to reflect your current setup of entities in your Local Identity Directory. For more information, see:

- [Manage Transformations \[page 1494\]](#)
- SCIM API version 2: [Identity Directory SCIM API](#) 

#### **Default transformation for Local Identity Directory:**

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem",
        "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ][ 'userId' ]"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.name.honorificPrefix",
```

```

        "targetPath": "$.name.honorificPrefix",
        "optional": true
    },
    {
        "sourcePath": "$.emails[*].value",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails[?(@.value)]"
    },
    {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active"
    },
    {
        "sourcePath": "$.userType",
        "targetPath": "$.userType",
        "optional": true
    },
    {
        "sourcePath": "$.addresses",
        "targetPath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
    },
    {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.timezone",
        "targetPath": "$.timezone",
        "optional": true
    },
    {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
    },
    {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "type": "remove",
        "targetPath": "$.groups[*].display"
    },
    {
        "condition": "$.displayName EMPTY true",
        "type": "remove",
        "targetPath": "$.displayName"
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",

```

```

        "optional": true
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
        "optional": true
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystem']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystem']",
        "optional": true
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true
      },
    ],
  },

```

```

        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
            "optional": true
        },
        {
            "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']",
            "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']",
            "optional": true
        },
        {
            "sourcePath": "$.company",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
            "optional": true
        }
    ]
},
"group": {
    "mappings": [
        {
            "sourcePath": "$.id",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
            "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName"
        },
        {
            "sourcePath": "$.members",
            "targetPath": "$.members",
            "preserveArrayWithSingleElement": true,
            "optional": true
        }
    ]
}
}

```

5. Add a target system where to provision users and groups. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Identity Directory \[page 1567\]](#)

### 1.6.1.3 SAP Advanced Financial Closing

Follow this procedure to set up a SAP Advanced Financial Closing as a source system.

## Prerequisites

You have created an instance and generated a service key for the standard service plan of SAP Advanced Financial Closing. For more information, see: [How to Manage User Access Using the SCIM API Provided](#).

The service key contains the API URL and the OAuth credentials (`clientid` and `clientsecret`) under the `uaa` property.

## Context

SAP Advanced Financial Closing allows you to define, automate, process, and monitor the financial closing tasks for the entities of your organization. It is an SAP BTP application that runs in an SAP BTP subaccount.

You can use Identity Provisioning to configure SAP Advanced Financial Closing as a source system where you can read users, user groups and user roles from, and provision them to a target system.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

2. Add *SAP Advanced Financial Closing* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Enter the URL provided by the service key under <a href="#">endpoints</a> <a href="#">scim2</a> without adding the path information.  For example: <code>https://afc-production-afc-api.cfapps.eu10.hana.ondemand.com</code>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user ID provided by the service key under <a href="#">uaa</a> <a href="#">clientid</a> .
Password	(Credential) Enter the password provided by the service key under <a href="#">uaa</a> <a href="#">clientsecret</a> .
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL provided by the service key of your SAP Advanced Financial Closing instance. It follows the pattern: <code>&lt;uaa.url&gt;/oauth/token</code> , where: <ul style="list-style-type: none"> <li>• <code>&lt;uaa.url&gt;</code> is the URL provided by the service key under <a href="#">uaa</a> <a href="#">url</a>.</li> <li>• <code>/oauth/token</code> is the suffix you need to add.</li> </ul>

Property Name	Value
(Optional) <code>s4hana.afc.user.filter</code>	<p>When specified, only those SAP Advanced Financial Closing users matching the filter expression will be read.</p> <p>Supported operators: <b>eq</b> (equal), <b>sw</b> (starts with) and <b>co</b> (contains)</p> <p>For example:</p> <ul style="list-style-type: none"> <li><code>userName eq "Julie Armstrong"</code></li> <li><code>emails eq "julie.armstrong@example.com"</code></li> <li><code>name.familyName sw "A"</code></li> <li><code>name.givenName co "Ju"</code></li> </ul>
(Optional) <code>s4hana.afc.group.filter</code>	<p>When specified, only those SAP Advanced Financial Closing users matching the filter expression will be read.</p> <p>Supported operators: <b>eq</b> (equal), <b>sw</b> (starts with) and <b>co</b> (contains)</p> <p>For example: <code>displayName eq "Administrators"</code></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the SAP Advanced Financial Closing source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Advanced Financial Closing. For more information, see [Manage Transformations \[page 1494\]](#).

[How to Manage User Access Using the SCIM API Provided](#)

[SAP Business Accelerator Hub: SAP Advanced Financial Closing](#)

### **i** Note

When configuring SAP Advanced Financial Closing as a source system, there may be a mismatch between optional user attributes in the source system and required user attributes in the target system. For example, the email is an optional attribute in SAP Advanced Financial Closing but a required one in Identity Authentication. In such cases, you need to review and adapt your source and/or target system transformations and configurations to reflect your needs.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
```



```

        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true
    },
    {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
    },
    {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.name.formatted",
        "targetPath": "$.name.formatted",
        "optional": true
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
    },
    {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
    },
    {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "optional": true
    },
    {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
    },
    {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "functions": [

```

```

        {
          "condition": "'%s4hana.afc.group.prefix%' != 'null'",
          "function": "concatString",
          "applyOnElements": true,
          "applyOnAttribute": "display",
          "prefix": "%s4hana.afc.group.prefix%"
        }
      ]
    },
    {
      "sourcePath": "$.roles",
      "targetPath": "$.roles",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "sourcePath": "$.phoneNumbers",
      "targetPath": "$.phoneNumbers",
      "preserveArrayWithSingleElement": true,
      "optional": true
    }
  ]
},
"group": {
  "mappings": [
    {
      "sourcePath": "$.id",
      "targetVariable": "entityIdSourceSystem"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "functions": [
        {
          "condition": "'%s4hana.afc.group.prefix%' != 'null'",
          "function": "concatString",
          "prefix": "%s4hana.afc.group.prefix%"
        }
      ]
    }
  ]
},
{
  "sourcePath": "$.members",
  "targetPath": "$.members",
  "preserveArrayWithSingleElement": true,
  "optional": true
}
]
}
}

```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.4 SAP Advanced Workflow

Follow this procedure to set up a source connector for SAP Advanced Workflow.

### Prerequisites

- You have technical credentials for SAP Advanced Workflow. Note that SAP Advanced Workflow is available to SAP Commissions customers as an optional add-on. You create an Admin user in SAP Commissions, which is synchronized with SAP Advanced Workflow. For more information, see: [Adding an Admin User](#) and [Commissions User Synchronization](#).
- You have set up SSO between Identity Authentication and SAP Advanced Workflow. For more information, see [Integration with SAP IdP](#).

### Context

SAP Advanced Workflow enables you to analyse, organize, and execute business processes to connect people, data, and daily activities. Workflow provides you the tools you need to configure and customize your business processes based on your specific business needs.

After fulfilling the prerequisites, follow the procedure below to add a source system for SAP Advanced Workflow where you can read **users** from and provision them to a target system. This source system consumes Workflow SCIM API.

#### i Note

SAP Advanced Workflow does not support groups.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Advanced Workflow* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

#### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your SAP Advanced Workflow system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the user for your SAP Advanced Workflow system.
Password	(Credential) Enter the password for your SAP Advanced Workflow user.
awf.domain	The domain name is the name of your SAP Advanced Workflow tenant.  If you don't know your tenant name, contact your supervisor or administrator, or refer to the email notification you received when your account was created.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Advanced Workflow](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SCIM system. For more information, see:

[Manage Transformations \[page 1494\]](#)

The behavior of the default transformation logic is to read all user attributes from the source SAP Advanced Workflow system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
```

```

        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
    },
    {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true
    },
    {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
    },
    {
        "sourcePath": "$.name.middleName",
        "optional": true,
        "targetPath": "$.name.middleName"
    },
    {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.name.familyName"
    },
    {
        "sourcePath": "$.phoneNumbers[0].value",
        "optional": true,
        "targetPath": "$.phoneNumbers[0].value"
    },
    {
        "sourcePath": "$.profileUrl",
        "targetPath": "$.profileUrl",
        "optional": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
    },
    {
        "sourcePath": "$.emails[0].value",
        "optional": true,
        "targetPath": "$.emails[0].value"
    },
    {
        "sourcePath": "$.timezone",
        "targetPath": "$.timezone",
        "optional": true
    }
]
}

```

5. Now, add a target system to provision users and their group assignments to it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.5 SAP Analytics Cloud

Follow this procedure to set up SAP Analytics Cloud as a source system.

### Prerequisites

- In SAP Analytics Cloud, you have enabled a custom SAML Identity Provider, for which *User Attribute* is set to **Custom SAML User Mapping**. To learn how, see: [Enabling a Custom SAML Identity Provider](#)
- Add an OAuth client with authorization grant **Client Credentials**. To learn how, see: [Managing OAuth Clients and Trusted Identity Providers](#)

### Context

SAP Analytics Cloud is an all-in-one cloud product offered as software as a service for business intelligence, planning, and predictive analytics.

You can use Identity Provisioning to configure SAP Analytics Cloud as a source system where you can read users and groups and provision them to target systems of your choice. The source system consumes SCIM 2.0 API provided by SAP Analytics Cloud.

There are two versions of the SAP Analytics Cloud SCIM API. They are handled by the `sac.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on April 10, 2023), SAP Analytics Cloud SCIM API version 1 is used. This is the default value.
- When the value is set to **2** - SAP Analytics Cloud SCIM API version 2 is used. This version is released with enhancements, such as: reading groups is not ignored in the default read transformation.

For more information on the differences between SAP Analytics Cloud SCIM API version 1 and 2, see [Managing Users and Teams](#).

For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Analytics Cloud* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter the URL to your SAP Analytics Cloud system without adding the path information.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the client ID to retrieve the OAuth access token for SAP Analytics Cloud.
Password	(Credential) Enter the client secret to retrieve the OAuth access token for SAP Analytics Cloud.
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Analytics Cloud instance.  This token URL is listed in the <i>OAuth Clients</i> section of the <i>App Integration</i> page. For more information, refer to <i>Authorize API Access for OAuth Clients</i> in <a href="#">Manage OAuth Clients</a>

Property Name	Description & Value
(Optional) <code>sac.api.version</code>	<p>Handles the version of SAP Analytics Cloud SCIM API.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">1</a> - Indicates that SAP Analytics Cloud SCIM API version 1 is used.</li> <li>• <a href="#">2</a> - Indicates that SAP Analytics Cloud SCIM API version 2 is used.</li> </ul> <p>Default value: <a href="#">1</a></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Analytic Cloud](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Analytic Cloud system. For more information, see: [Manage Transformations \[page 1494\]](#)

[SAP Analytics Cloud: User and Team Provisioning API](#)

[Managing Users and Teams → api/v1/scim](#)

[Managing Users and Teams → api/v1/scim2](#)

#### Default transformation for SAP Analytic Cloud SCIM API version 1:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName"
      }
    ]
  }
}
```



```

    {
      "sourcePath": "$.active",
      "targetPath": "$.active"
    },
    {
      "sourcePath": "$.emails",
      "targetPath": "$.emails",
      "preserveArrayWithSingleElement": true
    },
    {
      "sourcePath": "$.emails[?(@.primary== true)].value",
      "correlationAttribute": true
    },
    {
      "sourcePath": "$.roles",
      "targetPath": "$.roles",
      "preserveArrayWithSingleElement": true
    },
    {
      "sourcePath": "$.groups",
      "targetPath": "$.groups",
      "preserveArrayWithSingleElement": true,
      "functions": [
        {
          "condition": "%sac.group.prefix% != 'null'",
          "function": "concatString",
          "applyOnElements": true,
          "applyOnAttribute": "display",
          "prefix": "%sac.group.prefix%"
        }
      ]
    },
    {
      "sourcePath": "$
['urn:scim:schemas:extension:enterprise:1.0']['manager']['managerId']",
      "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']"
    }
  ],
  "group": {
    "ignore": true,
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "%sac.group.prefix% != 'null'",
            "function": "concatString",
            "prefix": "%sac.group.prefix%"
          }
        ]
      }
    ]
  },
  {
    "sourcePath": "$.members",
    "targetPath": "$.members",
    "preserveArrayWithSingleElement": true
  }

```

```

    },
    {
      "sourcePath": "$.roles",
      "targetPath": "$.roles",
      "preserveArrayWithSingleElement": true
    }
  ]
}

```

## Default transformation for SAP Analytic Cloud SCIM API version 2:

### Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "optional": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$
[ 'urn:sap:params:scim:schemas:extension:sac:2.0:user-custom-parameters' ]",
        "targetPath": "$
[ 'urn:sap:params:scim:schemas:extension:sac:2.0:user-custom-parameters' ]"
      }
    ]
  }
}

```

```

        {
            "sourcePath": "$",
            ["urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"],
            "targetPath": "$",
            ["urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"],
            "optional": true
        },
        {
            "sourcePath": "$.roles",
            "targetPath": "$.roles",
            "preserveArrayWithSingleElement": true,
            "optional": true
        },
        {
            "sourcePath": "$.groups",
            "targetPath": "$.groups",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "functions": [
                {
                    "condition": "'%sac.group.prefix%' != 'null'",
                    "function": "concatString",
                    "applyOnElements": true,
                    "applyOnAttribute": "display",
                    "prefix": "%sac.group.prefix%"
                }
            ]
        }
    ]
},
"group": {
    "mappings": [
        {
            "sourcePath": "$.schemas",
            "targetPath": "$.schemas",
            "preserveArrayWithSingleElement": true
        },
        {
            "sourcePath": "$.id",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName",
            "functions": [
                {
                    "condition": "'%sac.group.prefix%' != 'null'",
                    "function": "concatString",
                    "prefix": "%sac.group.prefix%"
                }
            ]
        }
    ],
    {
        "sourcePath": "$.members",
        "targetPath": "$.members",
        "optional": true,
        "preserveArrayWithSingleElement": true
    },
    {
        "sourcePath": "$",
        ["urn:sap:params:scim:schemas:extension:sac:2.0:group-roles"],
        "targetPath": "$",
        ["urn:sap:params:scim:schemas:extension:sac:2.0:group-roles"],
        "optional": true
    },
    {
        "sourcePath": "$",
        ["urn:sap:params:scim:schemas:extension:sac:2.0:group-custom-parameters"],

```

```

        "targetPath": "$
    [ 'urn:sap:params:scim:schemas:extension:sac:2.0:group-custom-parameters' ]",
        "optional": true
    }
  ]
}

```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

### 1.6.1.6 SAP Application Server ABAP

Follow this procedure to set up SAP Application Server ABAP (AS ABAP) as a source system.

## Prerequisites

### Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)
- You have credentials of a technical user with read permissions for AS ABAP, which plays the role of a user data source. Via this user, the Identity Provisioning service will call the ABAP public API in order to execute a number of function modules. These function modules are listed in **step 1** from the procedure below.
- You have the following role, which provides all authorizations with read-only access to user data:  
**SAP\_BCJSF\_COMMUNICATION\_RO**  
For more information, see: [Configuring the UME to Use an AS ABAP as Data Source](#)

## Context

SAP Application Server ABAP (AS ABAP) offers a user store and user administration capabilities for maintaining users and their authorizations for AS ABAP applications. You can configure AS ABAP as a source system for your identity provisioning process, in the following cases:

- Use AS ABAP as a central store for the identity data of your business users.
- Reuse the permission model, implemented in your AS ABAP client, as a permission model for cloud applications. For example, you can provision roles and permission assignments to SAP BTP.

### Note

During a provisioning job ([Read](#) or [Resync](#)), only active ABAP users are read. That means, users that have been created before the job is started, and whose expiration date is after the end of the job.

## Procedure

1. Open the Cloud Connector to add an access control system mapping for **AS ABAP**. This is needed to allow the Identity Provisioning service to access AS ABAP as a back-end system on the intranet. To learn how, see: [Configure Access Control \(RFC\)](#)

Go to ► [Cloud To On-Premise](#) ► [Access Control](#) ► tab and select protocol [RFC SNC](#). Then, expose the following *exact names* as accessible resources:

- PRGN\_ROLE\_GETLIST
- BAPI\_USER\_GETLIST
- BAPI\_USER\_GET\_DETAIL
- BAPI\_USER\_CREATE1
- BAPI\_USER\_ACTGROUPS\_ASSIGN
- IDENTITY\_MODIFY
- BAPI\_USER\_DELETE
- PRGN\_ACTIVITY\_GROUPS\_LOAD\_RFC

2. Open SAP BTP cockpit, and in your Identity Provisioning subaccount create a destination for the AS ABAP system. To learn how, see: [Create RFC Destinations](#)

The destination configuration is required by the Identity Provisioning service to find the back-end system to be used for reading data. It also provides the credentials of the technical user, needed for the connection to the ABAP public API.

Below are the fields you have to fill in the cockpit destination before using an AS ABAP client as a source system:

Field/Property Name	Value
<a href="#">Name</a>	Enter a destination name.

Field/Property Name	Value
<i>Type</i>	Select <a href="#">RFC</a> .
<i>User</i>	Enter the user for AS ABAP.  The <a href="#">User</a> field corresponds to property <code>jco.client.user</code> in the exported RFC destination.
<i>Password</i>	(Credential) Enter the password for the AS ABAP user.  The <a href="#">Password</a> field corresponds to property <code>jco.client.passwd</code> in the exported RFC destination.
<b>jco.client.client</b>	Provide the client to be used in the ABAP system. Valid format is a three-digit number.
<b>jco.destination.proxy_type</b>	Defines the proxy type of the connection you need to provide for your ABAP system.  The proxy type <a href="#">OnPremise</a> requires the Cloud Connector to access resources within your on-premise network.  Enter: <a href="#">OnPremise</a>
<b>Direct Connection</b>	
<b>jco.client.ashost</b>	Provide the virtual host entry that you have configured in the Cloud Connector → <a href="#">Access Control</a> configuration.
<b>jco.client.sysnr</b>	Provide the "system number" of the ABAP system.
<b>Load Balancing Connection</b>	
<b>jco.client.mshost</b>	Represents the message server host to be used.
<b>jco.client.r3name</b>	Provide the three-character system ID of the ABAP system to be addressed.
<b>jco.client.msservt</b>	Provide the port on which the message server is listening for incoming requests. You can use this property as an alternative to <code>jco.client.r3name</code> .
<b>Optional Properties</b>	
<b>jco.destination.peak_limit</b>	The value represents the maximum number of active connections that can simultaneously be created for a destination. For example: <a href="#">10</a>
<b>jco.destination.pool_capacity</b>	The value represents the maximum number of idle connections kept open by the destination. For example: <a href="#">5</a>

Field/Property Name	Value
<code>abap.user.name.filter</code>	<p>Filters user names by a regular expression. The regex can define any kind of search pattern.</p> <p>For example, <code>abap.user.name.filter = ^MAR.*</code> reads all user names that start with <i>MAR</i>, such as <b>MARK</b>, <b>MARTINA</b>, and so on.</p> <div> <p><b>i Note</b></p> <p>This property has a higher priority over <code>abap.user.filter</code>. That means, if you set both properties in a system, the value of <b><code>abap.user.name.filter</code></b> will be used. However, if the value of <code>abap.user.name.filter</code> is empty, then <b><code>abap.user.filter</code></b>'s value will be used instead.</p> </div>
<code>abap.role.name.filter</code>	<p>Filters user roles by a regular expression. The regex can define any kind of search pattern.</p> <p>For example, <code>abap.role.name.filter = ^inter.*</code> reads all users who have roles which start with <i>inter</i>, such as <b>internal</b>, <b>internship</b>, and so on.</p> <div> <p><b>i Note</b></p> <p>This property has a higher priority over <code>abap.role.filter</code>. That means, if you set both properties in a system, the value of <b><code>abap.role.name.filter</code></b> will be used. However, if the value of <code>abap.role.name.filter</code> is empty, then <b><code>abap.role.filter</code></b>'s value will be used instead.</p> </div>
<code>abap.user.membership.filter</code>	<p>Filters users by a regular expression, based on their <i>Role</i> memberships in AS ABAP. The regex can define any kind of search pattern.</p> <p>For example, <code>abap.user.membership.filter = (?i)^new.*</code> reads all users who have an assigned role which starts with <i>new</i>. This regex is case insensitive, which means the result can be roles starting with <b>new</b>, or <b>New</b>, or <b>NEW</b>, and so on.</p> <div> <p><b>i Note</b></p> <p>If connection properties, like <code>User</code> and <code>Password</code>, are configured both in the destination (SAP BTP cockpit) and on the <i>Properties</i> tab (Identity Provisioning User Interface), the values set in the destination are considered with higher priority.</p> </div>

3. Access Identity Provisioning. See: [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

4. Add *SAP Application Server ABAP* as a source system. To learn how, see: [Add a System \[page 1477\]](#)
5. From the *Destination Name* dropdown, choose the RFC destination you have created in [step 2](#).
6. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Application Server ABAP* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in AS ABAP. For more information, see [Manage Transformations \[page 1494\]](#).

When AS ABAP is configured as a source system for the Identity Provisioning service, the ABAP public API is used to retrieve the identity data from the AS ABAP system. During the reading process, the JSON data generated by the Identity Provisioning service, is following the structure of the API export parameters list and tables. Every BAPI table is represented as a JSON array and every BAPI structure is represented as a child JSON object.

#### Default transformation:

##### Code Syntax

```
// The value of attribute entityIdSourceSystem stores the unique ID of the
// identity. Do not delete this statement!
// You can exchange the default attribute USERNAME (which is used as a
// source) with another one, but make sure it is unique.
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.USERNAME",
        "targetVariable": "entityIdSourceSystem"
      },
    ],
  },

  // The USERNAME attribute is also used as userName value for the internal
  // JSON representation.
  {
    "sourcePath": "$.USERNAME",
    "targetPath": "$.userName",
    "correlationAttribute": true
  },
  {
    "sourcePath": "$.ALIAS.USERALIAS",
    "optional": true,
    "targetPath": "$.externalId",
    "correlationAttribute": true
  },
],

// The constant urn:ietf:params:scim:schemas:core:2.0:User is required as
// a value for the
// schemas definition in the Identity Authentication SCIM REST API.
{
  "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
  "targetPath": "$.schemas[0]"
},

// The ADDRESS.E_MAIL attribute is used also as a first array value in the
// emails JSON array.
{
  "sourcePath": "$.ADDRESS.E_MAIL",
  "optional": true,
  "targetPath": "$.emails[0].value",
```



```

    "correlationAttribute": true
  },
  {
    "condition": "$.ADDRESS.E_MAIL EMPTY false",
    "constant": true,
    "targetPath": "$.emails[0].primary"
  },
  {
    "condition": "$.ADDRESS.E_MAIL EMPTY false",
    "constant": "work",
    "targetPath": "$.emails[0].type"
  },
  {
    "sourcePath": "$.ADDRESS.FIRSTNAME",
    "optional": true,
    "targetPath": "$.name.givenName"
  },
  {
    "sourcePath": "$.ADDRESS.LASTNAME",
    "optional": true,
    "targetPath": "$.name.familyName"
  },
  {
    "sourcePath": "$.ADDRESS.MIDDLENAME",
    "optional": true,
    "targetPath": "$.name.middleName"
  },
  {
    "sourcePath": "$.ADDRESS.NICKNAME",
    "optional": true,
    "targetPath": "$.nickName"
  },
  {
    "sourcePath": "$.ADDRESS.TITLE_P",
    "optional": true,
    "targetPath": "$.name.honorificPrefix"
  },
  {
    "sourcePath": "$.ADDRESS.COUNTRY",
    "optional": true,
    "targetPath": "$.addresses[0].country"
  },
  {
    "condition": "$.ADDRESS.COUNTRY EMPTY false",
    "constant": true,
    "targetPath": "$.addresses[0].primary"
  },
  {
    "condition": "$.ADDRESS.COUNTRY EMPTY false",
    "constant": "work",
    "targetPath": "$.addresses[0].type"
  },
  {
    "sourcePath": "$.ADDRESS.TEL1_NUMBR",
    "optional": true,
    "targetPath": "$.phoneNumbers[0].value"
  },
  {
    "condition": "$.ADDRESS.TEL1_NUMBR EMPTY false",
    "constant": true,
    "targetPath": "$.phoneNumbers[0].primary"
  },
  {
    "condition": "$.ADDRESS.TEL1_NUMBR EMPTY false",
    "constant": "work",
    "targetPath": "$.phoneNumbers[0].type"
  },
}

```

```

// The Identity Provisioning reads the specific ABAP language codes and
mapped them as locales in the target system.
// The transformation provides an example with key = "W", which in the
target system is mapped as "bg". The default language is en.
// To see all languages and codes supported by AS ABAP, see the Related
Information section below.
{
  "optional": true,
  "targetPath": "$.locale",
  "type": "valueMapping",
  "sourcePaths": [
    "$.DEFAULTS.LANGU"
  ],
  "defaultValue": "en",
  "valueMappings": [
    {
      "key": [
        "W"
      ],
      "mappedValue": "bg"
    }
  ]
},
{
  "optional": true,
  "targetPath": "$.preferredLanguage",
  "type": "valueMapping",
  "functions": [
    {
      "function": "toLowerCaseString"
    }
  ],
  "sourcePaths": [
    "$.ADDRESS.LANGUP_ISO"
  ]
},
// The Identity Provisioning reads standard timezone codes, which are
supported by the AS ABAP BAPI.
// However, the standard SCIM API does not support these codes, thus the
target system can only accept values in format "<region>/<city>".
// The transformation provides an example with key = "EET", which in the
target system is mapped as "Europe/Sofia". The default timezone is Berlin.
{
  "optional": true,
  "targetPath": "$.timezone",
  "type": "valueMapping",
  "sourcePaths": [
    "$.LOGONDATA.TZONE"
  ],
  "defaultValue": "Europe/Berlin",
  "valueMappings": [
    {
      "key": [
        "EET"
      ],
      "mappedValue": "Europe/Sofia"
    }
  ]
},
{
  "constant": false,
  "targetPath": "$.active"
},
{
  "condition": "($.ISLOCKED.LOCAL_LOCK != 'L') &&
($.ISLOCKED.NO_USER_PW != 'L') && ($.ISLOCKED.GLOB_LOCK != 'L') &&
($.ISLOCKED.WRNG_LOGON != 'L')",
  "constant": true,

```

```

        "targetPath": "$.active"
    },
    // The attribute ACTIVITYGROUPS (SAP ABAP roles) is transformed by default
    // into groups attribute of the SCIM internal representation.
    {
        "sourcePath": "$.ACTIVITYGROUPS[*].AGR_NAME",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups[?(@.value)]"
    }
]
},
"group": {
    "mappings": [
        {
            "sourcePath": "$.ROLE_NAME",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "sourcePath": "$.ROLE_NAME",
            "targetPath": "$.displayName"
        },
        {
            "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath": "$.schemas[0]"
        },
        {
            "sourcePath": "$.USERLIST[*].USERNAME",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.members[?(@.value)]"
        }
    ]
}
}
}

```

#### Default transformation supporting User UUID attribute:

##### Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.USERNAME",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.USERNAME",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.ALIAS.USERALIAS",
        "optional": true,
        "targetPath": "$.externalId",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.SAPUSER_UUID.SAP_UID",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]"

```

```

    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": $.ADDRESS.E_MAIL,
      "optional": true,
      "targetPath": $.emails[0].value,
      "correlationAttribute": true
    },
    {
      "condition": $.ADDRESS.E_MAIL EMPTY false,
      "constant": true,
      "targetPath": $.emails[0].primary
    },
    {
      "condition": $.ADDRESS.E_MAIL EMPTY false,
      "constant": "work",
      "targetPath": $.emails[0].type
    },
    {
      "sourcePath": $.ADDRESS.FIRSTNAME,
      "optional": true,
      "targetPath": $.name.givenName
    },
    {
      "sourcePath": $.ADDRESS.LASTNAME,
      "targetPath": $.name.familyName
    },
    {
      "sourcePath": $.ADDRESS.MIDDLENAME,
      "optional": true,
      "targetPath": $.name.middleName
    },
    {
      "sourcePath": $.ADDRESS.NICKNAME,
      "optional": true,
      "targetPath": $.nickName
    },
    {
      "sourcePath": $.ADDRESS.TITLE_P,
      "optional": true,
      "targetPath": $.name.honorificPrefix
    },
    {
      "sourcePath": $.ADDRESS.COUNTRY,
      "optional": true,
      "targetPath": $.addresses[0].country
    },
    {
      "condition": $.ADDRESS.COUNTRY EMPTY false,
      "constant": true,
      "targetPath": $.addresses[0].primary
    },
    {
      "condition": $.ADDRESS.COUNTRY EMPTY false,
      "constant": "work",
      "targetPath": $.addresses[0].type
    },
    {
      "sourcePath": $.ADDRESS.TEL1_NUMBR,
      "optional": true,
      "targetPath": $.phoneNumbers[0].value
    },
    {
      "condition": $.ADDRESS.TEL1_NUMBR EMPTY false,
      "constant": true,

```

```

        "targetPath": "$.phoneNumbers[0].primary"
    },
    {
        "condition": "$.ADDRESS.TEL1_NUMBR EMPTY false",
        "constant": "work",
        "targetPath": "$.phoneNumbers[0].type"
    },
    {
        "type": "valueMapping",
        "sourcePaths": [
            "$.DEFAULTS.LANGU"
        ],
        "optional": true,
        "targetPath": "$.locale",
        "defaultValue": "en",
        "valueMappings": [
            {
                "key": [
                    "W"
                ],
                "mappedValue": "bg"
            }
        ]
    },
    {
        "type": "valueMapping",
        "sourcePaths": [
            "$.ADDRESS.LANGUP_ISO"
        ],
        "optional": true,
        "targetPath": "$.preferredLanguage",
        "functions": [
            {
                "function": "toLowerCaseString"
            }
        ]
    },
    {
        "type": "valueMapping",
        "sourcePaths": [
            "$.LOGONDATA.TZONE"
        ],
        "optional": true,
        "targetPath": "$.timezone",
        "defaultValue": "Europe/Berlin",
        "valueMappings": [
            {
                "key": [
                    "EET"
                ],
                "mappedValue": "Europe/Sofia"
            }
        ]
    },
    {
        "constant": false,
        "targetPath": "$.active"
    },
    {
        "condition": "($.ISLOCKED.LOCAL_LOCK != 'L') && ($.ISLOCKED.GLOB_LOCK != 'L') && ($.ISLOCKED.WRNG_LOGON != 'L')",
        "constant": true,
        "targetPath": "$.active"
    },
    {
        "sourcePath": "$.ACTIVITYGROUPS[*].AGR_NAME",
        "preserveArrayWithSingleElement": true,
        "optional": true,

```

```

        "targetPath": "$.groups[?(@.value)]"
      }
    ],
    },
    "group": {
      "mappings": [
        {
          "sourcePath": "$.ROLE_NAME",
          "targetVariable": "entityIdSourceSystem"
        },
        {
          "sourcePath": "$.ROLE_NAME",
          "targetPath": "$.displayName",
          "functions": [
            {
              "condition": "'%abap.role.prefix%' != 'null'",
              "function": "concatString",
              "prefix": "%abap.role.prefix%"
            }
          ]
        }
      ]
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.USERLIST[*].USERNAME",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.members[?(@.value)]",
      "optional": true
    }
  ]
}

```

7. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Example

### How to transform roles, assigned to AS ABAP users, into corporate groups in the Identity Authentication?

The AS ABAP roles are represented as groups in your Identity Authentication tenant. That is, when you configure AS ABAP as a source and Identity Authentication as a target system, the default transformations helps you to use the ABAP roles assignment of the users as source data and to automatically create corporate group assignments for the users in the Identity Authentication. When a user is assigned to one or more AS ABAP roles, the technical names of these roles (their ABAP attribute name is **AGR\_NAME**) will become corporate groups value in the Identity Authentication.

1. Transforming source data into the intermediate JSON representation.

The following example demonstrates how the sample roles, read from the AS ABAP system, will become groups in the intermediate JSON data, as a result from the transformation statement:

Data read from AS ABAP user store	Intermediate JSON data
<pre> Sample Code  ... "ACTIVITYGROUPS": [   {     "AGR_TEXT": "FICO 03",     "AGR_NAME": "ZFICO_03",     "FROM_DAT": "27.04.2016",     "TO_DAT": "31.12.9999"   },   {     "AGR_TEXT": "CASH 01",     "AGR_NAME": "ZCASH_01",     "FROM_DAT": "16.05.2016",     "TO_DAT": "31.12.9999"   } ] ... </pre>	<pre> Sample Code  ... "groups": [   {     "value": "ZFICO_03"   },   {     "value": "ZCASH_01"   },   ... ] </pre>

- The mapping statement in the default transformation, available when the Identity Authentication service is configured as a target system:

<pre> Sample Code  {   "sourcePath": "\$.groups",   "preserveArrayWithSingleElement": true,   "optional": true,   "targetPath": "\$.corporateGroups" } </pre>
---

- The following example demonstrates how the groups from the intermediate JSON are transformed into corporate groups, using the transformation statement:

Intermediate JSON Data	Transformation output result
<pre> Sample Code  ... "groups": [   {     "value": "ZFICO_03"   },   {     "value": "ZCASH_01"   },   ... ] </pre>	<pre> Sample Code  ... "corporateGroups": [   {     "value": "ZFICO_03"   },   {     "value": "ZCASH_01"   },   ... ] </pre>

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[ABAP: Supported Languages and Code Pages](#)

### 1.6.1.7 SAP Ariba Applications

Follow this procedure to set up SAP Ariba Applications as a source system.


## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

1. You have created a client application on [SAP Ariba APIs Portal](#)  that needs to be enabled for Identity Provisioning.

### i Note

If you don't have an account on SAP Ariba Developer Portal, then ask your **Designated Support Contact** (DSC) to submit a [request for an account](#). To find your DSC person, see: [How can I see my company's Basic users and Designated Support Contacts \(DSC\)](#) .

2. Provide your DSC person with your SAP Ariba **realm name**, **application name**, and **application key**. You have already created the application name along with the application key on [step 1](#). To find your realm name, login to your SAP Ariba system – it's part of your login URL, as shown in the following examples.
  - *SAP Ariba Buyer* example: <https://s1.ariba.com/Buyer/Main/ad/loginPage/...&realm=mycompany-t>
  - *SAP Ariba Sourcing* example: <http://mycompany.sourcing.ariba.com/>



3. Ask your DSC person to submit a service request for you to [SAP Ariba Support](#) for component **BNS-ARI-SS-API**, requesting the client application to be enabled for Identity Provisioning. Request your DSC person to mention the following details in the service request:
  - Application name
  - Application key
  - Realm name
4. When your application is enabled, you can login to [SAP Ariba APIs Portal](#), find your application, and generate a new OAuth secret for it. To learn how, see: [How to generate the OAuth Secret and Base64 Encoded Client and secret](#)
5. To configure your [SAP Ariba Applications](#) provisioning system (see the procedure below), you will need to map your SAP Ariba application parameters to the relevant Identity Provisioning properties. The property mapping between the two systems is as follows:

SAP Ariba	Identity Provisioning	Values
SCIM API URL	URL	Examples: <ul style="list-style-type: none"> <li>• US: <a href="https://openapi.ariba.com">https://openapi.ariba.com</a></li> <li>• Europe: <a href="https://eu.openapi.ariba.com">https://eu.openapi.ariba.com</a></li> <li>• UAE: <a href="https://mn1.openapi.ariba.com">https://mn1.openapi.ariba.com</a></li> </ul>
SAP Ariba OAuth 2.0 Token URL	OAuth2TokenServiceURL	Examples: <ul style="list-style-type: none"> <li>• US: <a href="https://api.ariba.com/v2/oauth/token">https://api.ariba.com/v2/oauth/token</a></li> <li>• Europe: <a href="https://api-eu.ariba.com/v2/oauth/token">https://api-eu.ariba.com/v2/oauth/token</a></li> <li>• UAE: <a href="https://api.mn1.ariba.com/v2/oauth/token">https://api.mn1.ariba.com/v2/oauth/token</a></li> </ul>
OAuth Client ID	User	Alphanumeric string Example: <b>aaaa12345-1111-3333-cccc-1234567890</b>
OAuth Secret	Password	Alphanumeric string Example: <b>aaaGGG1eee12abcdefGHIJK123lmnopTTT</b>
Application key	<code>ariba.applications.api.key</code>	Alphanumeric string Example: <b>123abc123XYZ000abc123ABC012345</b>
AN-ID	<code>ariba.applications.realm.id</code>	AN<numeric_string> Example: <b>AN000111222333</b>

## Context

You can read users and groups from SAP Ariba Applications source system and provision them to a target system of your choice.

These source systems consume SCIM 2.0 API provided by SAP Ariba Applications. For more information about the SAP Ariba SCIM API scope of support, see [3228340](#) .

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Ariba Applications* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter the SCIM API URL for your SAP Ariba application (see the <b>Prerequisites</b> section).
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client ID (see the <b>Prerequisites</b> section).
Password	Enter the OAuth Secret (see the <b>Prerequisites</b> section).
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL (see the <b>Prerequisites</b> section).
ariba.applications.api.key	Enter your application key (see the <b>Prerequisites</b> section).

Property Name	Description & Value
<code>ariba.applications.realm.id</code>	Enter your AN-ID (see the <b>Prerequisites</b> section).
(Optional) <code>ariba.applications.group.flatten</code>	<p>This property allows or forbids reading "nested groups" (group structures) from SAP Ariba Applications. If enabled (<b>true</b>), group members of type <i>group</i> will be ignored during read in order not to be provisioned to target systems that do not support nested groups. Thus, set it to <b>false</b> only if you are sure that the target system supports nested groups.</p> <p>Default value: <i>false</i></p> <p>Predefined value (during system creation): <i>true</i></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination (property configuration):

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://openapi.ariba.com
User=aaaa12345-1111-3333-cccc-1234567890
Password=*****
OAuth2TokenServiceURL=https://api.ariba.com/v2/oauth/token
ariba.applications.group.flatten=true
ariba.applications.api.key=123abc123XYZ000abc123ABC012345
ariba.applications.realm.id=AN000111222333
```

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Ariba Applications* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Ariba Applications source system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Ariba APIs Portal](#) → *Discover* → *SUPPLIER MANAGEMENT*

**Default transformation:**

## Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.emails[?(@.primary == true)].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]['userUuid']",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]['userUuid']",
        "optional": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.title",
        "targetPath": "$.title",
        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      },
      {
        "sourcePath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.phoneNumbers"
      }
    ]
  }
}
```

```

        {
            "sourcePath": "$.groups",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.groups",
            "functions": [
                {
                    "condition": "'%ariba.applications.group.prefix%' !"
                }
            ]
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
            "optional": true
        }
    ],
    "group": {
        "mappings": [
            {
                "sourcePath": "$.id",
                "targetVariable": "entityIdSourceSystem"
            },
            {
                "sourcePath": "$.schemas",
                "preserveArrayWithSingleElement": true,
                "targetPath": "$.schemas"
            },
            {
                "sourcePath": "$.displayName",
                "targetPath": "$.displayName",
                "functions": [
                    {
                        "condition": "'%ariba.applications.group.prefix%' !"
                    }
                ]
            }
        ]
    },
    {
        "sourcePath": "$.members",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members"
    }
]
}

```

- Now, add a target system to provision users into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[The SAP Ariba developer portal](#)

[Video: Create application and API approval process](#) 📺

### 1.6.1.8 SAP BTP ABAP environment

Follow this procedure to set up SAP BTP ABAP environment as a source system.

## Prerequisites

To establish the connection between Identity Provisioning and SAP BTP ABAP environment, you need to set up the communication (user, system and arrangement) on SAP BTP ABAP environment. You can do it now (as a prerequisite) or in the process of configuring SAP BTP ABAP environment as a source system, as described in step 3.

## Context

You can use SAP BTP ABAP environment to read entities from it and provision them to a target system. This scenario supports reading **business users** (Employee), **user assignments**, and **business roles** (which are considered as *groups*).


## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP BTP ABAP environment* as a source system. For more information, see [Add a System \[page 1477\]](#).

3. Set up the communication between Identity Provisioning and SAP BTP ABAP environment and configure your authentication method (basic or certificate-based).

#### **i Note**

We recommend that you use certificate-based authentication.

- a. In your newly added SAP BTP ABAP environment source system, select the [Certificate](#) tab and choose [Generate](#) > [Download](#) , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP BTP ABAP environment backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide [User Name](#) and [Password](#).

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide [System ID](#), [System Name](#) and [Host Name](#).

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose [Scenario ID](#) SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

For more information, see [Maintain a Communication Arrangement for Inbound Communication](#) .

#### **i Note**

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

#### **i Note**

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the API URL to your SAP BTP ABAP environment system.</p> <p>You can take the URL from the communication scenario SAP_COM_0193.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"> <li>• <a href="#">BasicAuthentication</a></li> <li>• <a href="#">ClientCertificateAuthentication</a></li> </ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div> <p><b>! Restriction</b></p> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <p><b>! Restriction</b></p> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
a4c.skip.read.archived	<p>In the event of archived (disabled) entities in a source SAP BTP ABAP environment system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>



Property Name	Description & Value
(Optional) <code>a4c.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP BTP ABAP environment system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ <b>4.5 Filter System Query Option</b></p>
(Optional) <code>a4c.roles.page.size</code>	<p>Indicate how many business roles (considered as <i>groups</i>) per page to be read from your SAP BTP ABAP environment system.</p> <p>The value must be an integer number.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://12345-aaaaa-3333.abap.hana.ondemand.com
User=MyABAPEnvUser
Password=*****
ips.date.variable.format=yyyy-MM-dd
a4c.skip.read.archived=true
a4c.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true
a4c.roles.page.size=30
```

##### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP BTP ABAP environment](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in your SAP BTP ABAP environment. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

**Default transformation:**

## Code Syntax

```
{
  "user": {
    "condition": "($ validityPeriod.startDate <= '${currentDate}') &&
($ validityPeriod.endDate > '${currentDate}')",
    "mappings": [
      {
        "sourcePath": "$ personID",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$ personalInformation.firstName",
        "targetPath": "$ name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$ personalInformation.lastName",
        "targetPath": "$ name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$ personalInformation.middleName",
        "targetPath": "$ name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$ personalInformation.personFullName",
        "targetPath": "$ name.formatted",
        "optional": true
      },
      {
        "sourcePath": "$ user.userName",
        "targetPath": "$ userName",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "constant": true,
        "targetPath": "$ active"
      },
      {
        "condition": "$ user.lockedIndicator == 'true'",
        "constant": false,
        "targetPath": "$ active",
        "optional": true
      }
    ],
    // The following condition states that if a business user is outside its
    // validity period, it will be set as inactive.
    // That means, this user will not be able to log into the SAP BTP ABAP
    // environment system.
    {
      "condition": "($ user.validityPeriod.startDate > '${currentDate}')
|| ('${currentDate}' > $ user.validityPeriod.endDate)",
      "constant": false,
      "optional": true,
      "targetPath": "$ active"
    },
    {
      "sourcePath": "$ workplaceInformation.emailAddress",
      "targetPath": "$ emails[0].value",
      "optional": true,
      "correlationAttribute": true
    },
    {
      "sourcePath": "$ user.logonLanguageCode",
      "optional": true,

```

```

        "targetPath": "$.locale"
    },
    {
        "sourcePath": "$.PersonExternalID",
        "optional": true,
        "correlationAttribute": true
    },
    // The Identity Provisioning reads both users and user assignments from
    // SAP BTP ABAP environment.
    {
        "sourcePath": $.user.role,
        "optional": true,
        "targetPath": $.groups,
        "preserveArrayWithSingleElement": true,
        "functions": [
            {
                "condition": "%a4c.roles.prefix% != 'null'",
                "function": "concatString",
                "applyOnElements": true,
                "prefix": "%a4c.roles.prefix%",
                "applyOnAttribute": "roleName",
                "assignToAttribute": "display"
            },
            {
                "condition": "%a4c.roles.prefix% == 'null'",
                "function": "concatString",
                "applyOnElements": true,
                "prefix": "",
                "applyOnAttribute": "roleName",
                "assignToAttribute": "display"
            },
            {
                "function": "concatString",
                "applyOnElements": true,
                "prefix": "",
                "applyOnAttribute": "roleName",
                "assignToAttribute": "value"
            }
        ]
    },
    {
        "type": "remove",
        "targetPath": $.groups[*].roleName
    },
    {
        "sourcePath": $.user.globalUserID,
        "optional": true,
        "targetPath": $
    },
    ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']
    },
    {
        "type": "valueMapping",
        "sourcePaths": [
            $.user.timeZoneCode
        ],
        "targetPath": $.timezone,
        "defaultValue": "Europe/Berlin",
        "valueMappings": [
            {
                "key": [
                    "WDF"
                ],
                "mappedValue": "Europe/Berlin"
            },
            {
                "key": [
                    "ISRAEL"
                ],

```

```

    "mappedValue": "Asia/Jerusalem"
  },
  {
    "key": [
      "RUS03"
    ],
    "mappedValue": "Europe/Moscow"
  },
  {
    "key": [
      "AUSNSW"
    ],
    "mappedValue": "Australia/Sydney"
  },
  {
    "key": [
      "UTC+4"
    ],
    "mappedValue": "Asia/Dubai"
  },
  {
    "key": [
      "BRAZIL"
    ],
    "mappedValue": "America/Sao_Paulo"
  },
  {
    "key": [
      "BRZLEA"
    ],
    "mappedValue": "America/Sao_Paulo"
  },
  {
    "key": [
      "MSTNO"
    ],
    "mappedValue": "America/Phoenix"
  },
  {
    "key": [
      "EST"
    ],
    "mappedValue": "America/New_York"
  },
  {
    "key": [
      "UTC"
    ],
    "mappedValue": "Etc/UTC"
  },
  {
    "key": [
      "UTC+3"
    ],
    "mappedValue": "Asia/Riyadh"
  },
  {
    "key": [
      "EST_"
    ],
    "mappedValue": "America/Toronto"
  },
  {
    "key": [
      "UTC+8"
    ],
    "mappedValue": "Asia/Shanghai"
  },
  },

```

```

        {
          "key": [
            "JAPAN"
          ],
          "mappedValue": "Asia/Tokyo"
        }
      ],
    },
    {
      "type": "valueMapping",
      "sourcePaths": [
        "$.businessPartnerRoleCode"
      ],
      "targetPath": "$.userType",
      "defaultValue": "Employee",
      "valueMappings": [
        {
          "key": [
            "BUP003"
          ],
          "mappedValue": "Employee"
        }
      ]
    }
  ],
},
"group": {
  "mappings": [
    {
      "sourcePath": "$.ID",
      "targetVariable": "entityIdSourceSystem"
    },
    {
      "sourcePath": "$.ID",
      "functions": [
        {
          "condition": "'%a4c.roles.prefix%' != 'null'",
          "function": "concatString",
          "prefix": "%a4c.roles.prefix%"
        }
      ],
      "targetPath": "$.displayName"
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.to_BusinessUserAssignment.results",
      "optional": true,
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.members"
    },
    {
      "type": "remove",
      "targetPath": "$.members[*].__metadata"
    },
    {
      "type": "rename",
      "constant": "value",
      "targetPath": "$.members[*].PersonID"
    },
    {
      "constant": "User",
      "targetPath": "$.members[*].type"
    }
  ]
}
}

```

```
}
```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource. To learn how, see [Guided Answers: Business Role Description](#).

6. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA Cloud Documentation](#)

### 1.6.1.9 SAP BTP Account Members (Neo)

Follow this procedure to set up your SAP Business Technology Platform as a source system, from which you can read and manage members in your Neo account.

## Prerequisites

You have created a new Platform API OAuth client, with API *Account Member Management* and scopes *Manage Account Members* and *Read Account Members*.

Save the *Client ID* and *Client Secret* as you'll need them when you configure your source system. Make sure you save the client secret as you cannot retrieve it later.

For more information, see [Create a Platform API Client](#).

## Context

The Identity Provisioning service helps companies to automatically manage the user-to-platform roles assignments for SAP Business Technology Platform subaccounts. For this scenario, as a source system are

used SAP BTP subaccounts. The target system can be a solution supported by the Identity Provisioning service with write access for user and group artifacts.

This provisioning scenario is based on the Platform Authorization Management API of SAP BTP. For more information, see [Platform Authorization Management API](#).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP BTP Account Members (Neo)* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter: <b><code>https://api.&lt;SAP_BTP_host&gt;/authorization/v1/platform/accounts/&lt;SAP_BTP_account&gt;</code></b>  Examples: <ul style="list-style-type: none"><li>• (Europe – Rot) <code>https://api.hana.ondemand.com/authorization/v1/platform/accounts/abc123xyz</code></li><li>• (Japan – Tokyo) <code>https://api.jp1.hana.ondemand.com/authorization/v1/platform/accounts/abc123xyz</code></li></ul>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the Client ID of the new Platform API OAuth client created for the <b>Account Member Management</b> API (see the prerequisites).

Property Name	Description & Value
Password	Enter the Client Secret of the new Platform API OAuth client created for the <b>Account Member Management</b> API (see the prerequisites).
OAuth2TokenServiceURL	Enter: <b>https://api.&lt;SAP_BTP_host&gt;/oauth2/apitoken/v1</b> Examples: <ul style="list-style-type: none"> <li>(Europe – Rot) <a href="https://api.hana.ondemand.com/oauth2/apitoken/v1">https://api.hana.ondemand.com/oauth2/apitoken/v1</a></li> <li>(US East – Sterling) <a href="https://api.us3.hana.ondemand.com/oauth2/apitoken/v1">https://api.us3.hana.ondemand.com/oauth2/apitoken/v1</a></li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP BTP Account Members (Neo)* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

Using the default transformation, all users and platform roles available in the SAP Business Technology Platform source system will be created as users and groups in the chosen target system.

You can change the default transformation mapping rules to reflect the data to be written in your source system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP BTP: Authorization Management API](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
      }
    ]
  }
}
```



```

    },
    {
      "sourcePath": "$.name.familyName",
      "optional": true,
      "targetPath": "$.name.familyName"
    },
    {
      "sourcePath": "$.emails[0].value",
      "optional": true,
      "targetPath": "$.emails[0].value"
    },
    {
      "sourcePath": "$.description",
      "optional": true,
      "targetPath": "$.description"
    },
    {
      "sourcePath": "$.schemas",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.schemas"
    },
    {
      "sourcePath": "$.roles[*].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.groups[?(@.value)]"
    },
    {
      "type": "remove",
      "targetPath": "$.meta"
    }
  ]
},
"group": {
  "mappings": [
    {
      "sourcePath": "$.id",
      "targetPath": "$.id",
      "targetVariable": "entityIdSourceSystem"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "functions": [
        {
          "condition": "'%scp.group.prefix%' != 'null'",
          "function": "concatString",
          "prefix": "%scp.group.prefix%"
        }
      ]
    }
  ],
  {
    "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
    "targetPath": "$.schemas[0]"
  },
  {
    "sourcePath": "$.members[*].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members[?(@.value)]"
  },
  {
    "constant": "User",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members[*].type"
  }
]
}

```

```
}  
}
```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[What is SAP BTP](#)

### 1.6.1.10 SAP BTP Java/HTML5 apps (Neo)

Follow this procedure to set up SAP Business Technology Platform as a source system, from which you can read user-to-groups assignments for Java/HTML5 applications that run on SAP BTP, Neo environment.

## Prerequisites

You have created a new Platform API OAuth client, with [Authorization Management](#) scopes. Save the [Client ID](#) and [Client Secret](#) as you'll need them when you configure your source system. Make sure you save the client secret as you cannot retrieve it later.

For more information, see [Create a Platform API Client](#).

## Context

The Identity Provisioning service helps companies to automatically manage the user-to-groups assignments for Java/HTML5 applications running on SAP BTP, Neo environment. For this scenario, SAP BTP is the source system. The target system can be a solution supported by the Identity Provisioning service with write access for user and group artifacts.

This provisioning scenario is based on the Authorization Management REST API of the SAP Business Technology Platform. For more information, see [Using the Authorization Management REST API](#).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP BTP Java/HTML5 apps (Neo)* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter: <b>https://api.&lt;SAP_BTP_host&gt;/authorization/v1/accounts/&lt;SAP_BTP_account&gt;</b>  Examples: <ul style="list-style-type: none"><li>• (Europe – Rot) <i>https://api.hana.ondemand.com/authorization/v1/accounts/abc123xyz</i></li><li>• (Japan – Tokyo) <i>https://api.jp1.hana.ondemand.com/authorization/v1/accounts/abc123xyz</i></li></ul>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the Client ID of the new Platform API OAuth client created for the Authorization Management API (see the prerequisites).
Password	Enter the Client Secret of the new Platform API OAuth client created for the Authorization Management API (see the prerequisites).

Property Name	Description & Value
OAuth2TokenServiceURL	<p>Enter: <b>https://api.&lt;SAP_BTP_host&gt;/oauth2/apitoken/v1</b></p> <p>Examples:</p> <ul style="list-style-type: none"> <li>(Europe – Rot) <a href="https://api.hana.ondemand.com/oauth2/apitoken/v1">https://api.hana.ondemand.com/oauth2/apitoken/v1</a></li> <li>(US East – Sterling) <a href="https://api.us3.hana.ondemand.com/oauth2/apitoken/v1">https://api.us3.hana.ondemand.com/oauth2/apitoken/v1</a></li> </ul>
hcp.application.names	<p>Enter a comma-separated list of application names. That could be applications deployed on your account, or applications for which your account has subscribed. The property returns the roles assigned to these applications.</p> <p>Use the following format (no spaces):</p> <p><b>&lt;app_name1&gt;,&lt;app_name2&gt;,&lt;provider_subaccount&gt;:&lt;provider_app&gt;</b></p> <div> <p><b>⚠ Caution</b></p> <p>You must not leave this property with an empty value.</p> </div>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP BTP Java/HTML5 apps (Neo)* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

Using the default transformation, all groups and their members (with roles) available in the SAP Business Technology Platform source system will be created as groups and respective members (with roles) in the chosen target system.

You can change the default transformation mapping rules to reflect the data to be written in the target system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP BTP: Authorization Management API](#)

#### Default transformation:

##### Code Syntax

```
// To read the roles assigned to your applications, add property
hcp.application.names to your system configuration.
{
  "role": {
    "mappings": [
      {
```

```

        "targetPath": "$",
        "sourcePath": "$"
    },
    {
        "targetVariable": "entityIdSourceSystem",
        "constant": "-",
        "functions": [
            {
                "type": "concatString",
                "prefix": $.applicationName,
                "suffix": $.name
            }
        ]
    }
]
},
"group": {
    "mappings": [
        {
            "targetPath": $.id,
            "sourcePath": $.name
        },
        {
            "targetVariable": "entityIdSourceSystem",
            "sourcePath": $.name
        },
        {
            "targetPath": $.displayName,
            "sourcePath": $.name,
            "functions": [
                {
                    "condition": "'%hcp.group.prefix%' != 'null'",
                    "function": "concatString",
                    "prefix": "%hcp.group.prefix%"
                }
            ]
        }
    ],
    {
        "targetPath": $.schemas[0],
        "constant": "urn:ietf:params:scim:schemas:core:2.0:Group"
    },
    {
        "sourcePath": $.members,
        "preserveArrayWithSingleElement": true,
        "targetPath": $.members[?(@.value)],
        "optional": true
    },
    {
        "constant": "User",
        "preserveArrayWithSingleElement": true,
        "targetPath": $.members[*].type,
        "optional": true
    }
]
},
"user": {
    "mappings": [
        {
            "targetPath": $.id,
            "sourcePath": $.name
        },
        {
            "targetVariable": "entityIdSourceSystem",
            "sourcePath": $.name
        },
        {
            "targetPath": $.displayName,
            "sourcePath": $.name
        }
    ]
}

```

```

    },
    {
      "targetPath": "$.userName",
      "sourcePath": "$.name",
      "correlationAttribute": true
    },
    {
      "targetPath": "$.schemas[0]",
      "constant": "urn:ietf:params:scim:schemas:core:2.0:User"
    },
    {
      "sourcePath": "$.groups",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.groups[?(@.value)]",
      "optional": true
    },
    {
      "constant": "direct",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.groups[*].type",
      "optional": true
    }
  ]
}

```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[What is SAP BTP](#)

## 1.6.1.11 SAP BTP XS Advanced UAA (Cloud Foundry)

Follow this procedure to set up the SAP BTP XS Advanced UAA (running on SAP BTP, Cloud Foundry environment) as a source system.

### Prerequisites

- You have a technical database user for a SAP BTP XS Advanced UAA system with read access permissions. To learn how, see: [Get API Access](#)
- You have a subaccount on SAP BTP, Cloud Foundry environment, and Cloud Foundry applications for which you have been subscribed.
- Since OAuth is used for authentication of your service instance, you need to generate a service key for the service instance, and then retrieve this service key with OAuth 2.0 client credentials (client ID and secret). You'll use them when creating a destination (or specifying the Identity Provisioning connection properties) for access token retrieval. To learn how to generate XSUAA OAuth credentials, see: [Retrieve Credentials for Remote Applications](#)

### Context

In simple terms, XS Advanced is basically the Cloud Foundry open-source PaaS with a number of tweaks and extensions provided by SAP. These SAP enhancements include integration with the SAP HANA database, OData support, compatibility with XS classic model, and some additional features designed to improve application security. XS Advanced also provides support for business applications that are composed of multiple micro-services, also known as multi-target applications.

SAP BTP XS Advanced UAA is responsible for the connection of identity providers with business users (for applications). SAP BTP XS Advanced UAA provides authorizations on application level: *role collections*, *roles*, *attributes*, and *role templates*. To learn more, see: [What Is the SAP Authorization and Trust Management Service?](#)

Follow the steps below to create SAP BTP XS Advanced UAA as a source system to read SAP BTP users and groups from your Cloud Foundry applications, which can then be provisioned to a certain target system.

These source systems consume SCIM 1.1 API provided by SAP BTP XS Advanced UAA.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP BTP XS Advanced UAA (Cloud Foundry)* as a source system. For more information, see [Add a System \[page 1477\]](#).

3. Choose the [Properties](#) tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL to the SCIM API of your SAP BTP XS Advanced UAA system.</p> <p>If not sure about the exact URL, ask your SAP BTP XS UAA administrator.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
OAuth2TokenServiceURL	<p>As you need to make OAuth authentication to the SAP BTP XS Advanced UAA system, enter the URL to the OAuth2 token service</p> <p>If not sure about the exact URL, ask your SAP BTP XS UAA administrator.</p>
User	Enter the OAuth client ID of the SAP BTP XS Advanced UAA technical user (see <b>Prerequisites</b> ).
Password	Enter the OAuth client secret of the technical user (see <b>Prerequisites</b> ).
xsuaa.origin	<p>Enter the location of your identity provider. To do this:</p> <ol style="list-style-type: none"><li>1. Open your SAP BTP cockpit.</li><li>2. Go to your Cloud Foundry global account and choose your subaccount.</li><li>3. From the left-side navigation, choose <a href="#">Trust Configuration</a>.</li><li>4. Copy/paste the <a href="#">Origin Key</a> value.</li></ol> <p>This value will be used as the <a href="#">origin</a> attribute in the system transformation.</p>



Property Name	Description & Value
<code>xsuaa.origin.filter.enabled</code>	<p>This flag property depends on <code>xsuaa.origin</code>. Possible values: <b>true</b> or <b>false</b></p> <ul style="list-style-type: none"> <li>• If set to <b>true</b>, the Identity Provisioning service will read only users whose identity provider is set as a value of <code>xsuaa.origin</code>.</li> <li>• If set to <b>false</b>, the Identity Provisioning service will read all users, regardless of their origin.</li> <li>• If set to <b>true</b> but the <code>xsuaa.origin</code> property is missing, the provisioning job will fail.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

Type=[HTTP](#)

Authentication=[BasicAuthentication](#)

ProxyType=[Internet](#)

URL=<https://api.authentication.eu10.hana.ondemand.com>

OAuth2TokenServiceURL=<https://myaccount.authentication.eu10.hana.ondemand.com/oauth/token>

User=[MyXSUAAuser](#)

Password=\*\*\*\*\*

`xsuaa.origin`=[myaccount-xsuaa.accounts.ondemand.com](#)

`xsuaa.origin.filter.enabled`=[true](#)

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP BTP Advanced UAA](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

- **Mapping logic** – The behavior of the default transformation logic is to map all attributes from an SAP BTP XS Advanced UAA entity to the intermediate Identity Provisioning representation.
- **User offboarding** – If a user or group has been deleted from the SAP BTP XS Advanced UAA, this change is recognized and the user/group is deleted in the target system too.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP BTP XS Advanced UAA system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Cloud Foundry UAA API: Users](#) ➡

[Cloud Foundry UAA API: Groups](#) ➡

## Default transformation:

### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "functions": [
          {
            "condition": "'%xsuaa.group.prefix%' !== 'null'",
            "function": "concatString",
            "applyOnElements": true,
            "applyOnAttribute": "display",
            "prefix": "%xsuaa.group.prefix%"
          }
        ]
      },
      {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.verified",
        "targetPath": "$.verified",
        "optional": true
      },
      {
        "sourcePath": "$.meta",
        "targetPath": "$.meta",
        "optional": true
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath": "$.externalId",
      "targetPath": "$.externalId",
      "optional": true
    },
    {
      "sourcePath": "$.origin",
      "targetPath": "$.origin",
      "optional": true
    },
    {
      "sourcePath": "$.zoneId",
      "targetPath": "$.zoneId",
      "optional": true
    },
    {
      "sourceVariable": "entityBaseLocation",
      "targetPath": "$.meta.location",
      "targetVariable": "entityLocationSourceSystem",
      "functions": [
        {
          "type": "concatString",
          "suffix": "${entityIdSourceSystem}"
        }
      ]
    }
  ]
},
"group": {
  "mappings": [
    {
      "sourcePath": $.id,
      "targetPath": $.id,
      "targetVariable": "entityIdSourceSystem"
    },
    {
      "sourcePath": $.displayName,
      "targetPath": $.displayName,
      "functions": [
        {
          "condition": "'%xsuaa.group.prefix%' != 'null'",
          "function": "concatString",
          "prefix": "%xsuaa.group.prefix%"
        }
      ]
    }
  ]
},
{
  "sourcePath": $.description,
  "targetPath": $.description,
  "optional": true
},
{
  "sourcePath": $.zoneId,
  "targetPath": $.zoneId,
  "optional": true
},
{
  "sourcePath": $.meta,
  "targetPath": $.meta,
  "optional": true
},
{
  "sourcePath": $.members,
  "targetPath": $.members,
  "preserveArrayWithSingleElement": true,
  "optional": true
},

```

```

    {
      "sourceVariable": "entityBaseLocation",
      "targetPath": "$.meta.location",
      "targetVariable": "entityLocationSourceSystem",
      "functions": [
        {
          "function": "concatString",
          "suffix": "${entityIdSourceSystem}"
        }
      ]
    }
  ]
}

```

5. Now, add a target system to which to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[XS CLI: User Administration](#)

[Cloud Foundry UAA: Users](#) ➔

[Cloud Foundry UAA: Groups](#) ➔

### 1.6.1.11.1 Configure Single and Multiple Origins

Configure and provision users with single or multiple origins in SAP BTP XS Advanced UAA (Cloud Foundry) source system.

## Overview

An origin tells you which is the identity provider of a user in SAP BTP XS Advanced UAA (Cloud Foundry). It is defined in the trust configuration in the SAP BTP cockpit under [Security](#) ➤ [Trust Configuration](#) ➤ [Origin Key](#) ➤.

The origin itself is not a concept of Identity Provisioning. The role of the service is to ensure that you can read and provision users with their identity providers. Once you find the identity provider in the Origin Key, you need to set it in the `xsuaa.origin` property. You can configure it in source, target and proxy SAP BTP XS Advanced UAA (Cloud Foundry) systems. Both single and multiple values are supported. The value is a string that usually specifies the name of the identity provider or its location.

For example: `xsuaa.origin=ldap` and `xsuaa.origin=ldap;myaccount-xsuaa.accounts.ondemand.com`, where the ";" (semicolon) is the only supported delimiter.

## Provisioning Users with Single Origin

You want to provision a user with a single origin from SAP BTP XS Advanced UAA (Cloud Foundry) source system to a given target system.

1. On the [Properties](#) tab of the source system, set the `xsuaa.origin.filter.enabled` property to [true](#). This is a prerequisite for enabling the `xsuaa.origin` property in source systems.
2. Enter the value for the `xsuaa.origin` property, for example: [idp1](#). The value of this property acts like a filter as only users with the values specified there are read.
3. Run a provisioning job.

As a result, one user entry with single origin is read and created in the target system.

## Provisioning Users with Multiple Origins

You want to provision a user with multiple origins from SAP BTP XS Advanced UAA (Cloud Foundry) source system to a given target system.

1. On the [Properties](#) tab of the source system, set the `xsuaa.origin.filter.enabled` property to [true](#). This is a prerequisite for enabling the `xsuaa.origin` property in source systems.
2. Enter multiple values for the `xsuaa.origin` property, for example: [idp1;idp2](#). The value of this property acts like a filter as only users with the values specified there are read.
3. Set the `ips.delete.existedbefore.entities` property to [true](#) on the target system. This is needed before you run the provisioning job. It will ensure that previously existed users in the target system will be deleted if they were deleted in the source. For more information, see [Manage Deleted Entities \[page 1522\]](#)
4. Run a provisioning job.

As a result, two user entries for one and the same user are read. Only one user is created in the target system, because it does not support the origin concept and recognizes both user entries as one user. The job logs show that two users are read, one is created, and one is updated on the target system. A deletion of one of the origins in the source system would result in the following statistics: one user is read and one user is updated.

### Note

Multiple origins are not supported in provisioning scenarios between SAP BTP XS Advanced UAA (Cloud Foundry) source system and SAP BTP XS Advanced UAA (Cloud Foundry) target system.

## 1.6.1.12 SAP Build Work Zone, advanced edition

Follow this procedure to set up SAP Build Work Zone, advanced edition as a source system.

### Prerequisites

You have OAuth credentials for SAP Build Work Zone, advanced edition. To learn how, see [SAP Build Work Zone, advanced edition: Add an OAuth Client](#)

### Context

After fulfilling the prerequisites, follow the procedure below to create a source SAP Build Work Zone, advanced edition system to read users and groups.

These source systems consume SCIM 2.0 API provided by SAP Build Work Zone, advanced edition.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Build Work Zone, advanced edition* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

#### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>

Property Name	Description & Value
URL	Enter the URL related to your SAP Build Work Zone, advanced edition system, in format: <b>https://&lt;account&gt;&lt;sap_wz_domain&gt;.workzone.ondemand.com</b>  For example: <a href="https://mytenant.mydomain123.workzone.ondemand.com">https://mytenant.mydomain123.workzone.ondemand.com</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth client key, created for your SAP Build Work Zone, advanced edition tenant (see <b>Prerequisites</b> ).
Password	Enter the OAuth client secret, created for your SAP Build Work Zone, advanced edition tenant (see <b>Prerequisites</b> ).
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Build Work Zone, advanced edition instance, in format: <b>https://&lt;account&gt;&lt;sap_wz_domain&gt;.workzone.ondemand.com/api/v1/auth/token</b>  For example: <a href="https://myaccount.mydomain123.workzone.ondemand.com/api/v1/auth/token">https://myaccount.mydomain123.workzone.ondemand.com/api/v1/auth/token</a>
Optional Properties	
(Optional) <code>workzone.group.filter</code>	Enter a SCIM-based search criteria for filtering groups.  Example: <b>displayName eq "Project123"</b>
(Optional) <code>workzone.user.filter</code>	Enter a SCIM-based search criteria for filtering users.  Example: <b>userName eq "SmithJ" and addresses.country eq "US"</b>
(Optional) <code>ips.failed.request.retry.attempts</code>	Predefined value: <a href="#">2</a>
(Optional) <code>ips.failed.request.retry.attempts.interval</code>	Predefined value: <a href="#">30</a>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Build Work Zone, advanced edition* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Build Work Zone, advanced edition system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Build Work Zone OData API](#) 

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "type": "remove",
        "targetPath": "$.id"
      },
      {
        "type": "remove",
        "targetPath": "$.meta"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.title",
        "targetPath": "$.title",
        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      }
    ]
  }
}
```



```

    },
    {
      "sourcePath": "$.timezone",
      "optional": true,
      "targetPath": "$.timezone"
    },
    {
      "sourcePath": "$.addresses",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.addresses"
    },
    {
      "sourcePath": "$.groups",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.groups"
    },
    {
      "sourcePath": "$.schemas",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.schemas"
    },
    {
      "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
      "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
      "optional": true
    },
    {
      "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
      "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
      "optional": true
    },
    {
      "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
      "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
      "optional": true
    },
    {
      "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
      "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
      "optional": true
    },
    {
      "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
      "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
      "optional": true
    },
  },

```

```

        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
            "optional": true
        },
        {
            "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
            "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
            "optional": true
        },
        {
            "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
            "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
            "optional": true
        }
    ]
},
"group": {
    "ignore": true,
    "mappings": [
        {
            "sourcePath": "$.id",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName"
        },
        {
            "sourcePath": "$.members",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.members"
        },
        {
            "sourcePath": "$.schemas",
            "preserveArrayWithSingleElement": true,
            "targetPath": "$.schemas"
        },
        {
            "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
            "optional": true,
            "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']"
        },
        {
            "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['description']",

```

```

        "optional": true,
        "targetPath": "$
    [ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'description' ]"
    }
  ]
}

```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps


1. Before starting a provisioning job, you can first subscribe to the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during your jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.13 SAP Business Network

Follow this procedure to set up SAP Business Network as a source system.

### Context

#### i Note

Currently, SAP Business Network connector is only available for selected customers who are approached by SAP. For more information, see [3305074](#) 

SAP Business Network, formerly known as Ariba Network, is a cloud-based offering that makes it possible for buyers and suppliers to collaborate on transactions, strengthen their relationships, and discover new business opportunities.

You can use Identity Provisioning to configure SAP Business Network as a source system where you can read users and groups and provision them to target systems of your choice.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

2. Add *SAP Business Network* as a source system. See: [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the SAP Business Network API URL.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth Client Id, created for your SAP Business Network system.
Password	(Credential) Enter the OAuth Client Secret, created for your SAP Business Network system.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.
bn.api.key	An API Key represents the unique key that identifies a particular application as a legitimate consumer of an API.
bn.realm.id	The realm name is part of the URL you use to access SAP Business Network.
(Optional) bn.user.filter	<p>When specified, only those SAP Business Network users matching the filter expression will be read. For example:</p> <ul style="list-style-type: none"> <li>• <i>userName eq "Julie Armstrong"</i></li> <li>• <i>userName sw "J"</i></li> <li>• <i>emails eq "julie.armstrong@example.com"</i></li> </ul> <p>For more information, see <a href="#">List of Properties [page 94]</a></p>

Property Name	Value
(Optional) <code>bn.group.filter</code>	When specified, only those SAP Business Network groups matching the filter expression will be read. For example:  <i><code>displayName eq "Employees"</code></i>  For more information, see <a href="#">List of Properties [page 94]</a>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Business Network* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your system. For more information, see:

[Manage Transformations \[page 1494\]](#)

**Mapping logic** – the behavior of the default transformation logic is to read all user attributes from the source SAP Business Network system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ] [ 'userUuid' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ] [ 'userUuid' ]",
        "optional": true
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ] [ 'sendMail' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ] [ 'sendMail' ]",
        "optional": true
      }
    ]
  }
}
```

```

        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:sap.business.network:2.0:User']
['userStatus']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:sap.business.network:2.0:User']
['userStatus']",
        "optional": true
    },
    {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "correlationAttribute": true,
        "optional": true
    },
    {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName"
    },
    {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
    },
    {
        "sourcePath": "$.name.formatted",
        "targetPath": "$.name.formattedName",
        "optional": true
    },
    {
        "sourcePath": "$.name.honorificPrefix",
        "targetPath": "$.name.honorificPrefix",
        "optional": true
    },
    {
        "sourcePath": "$.name.honorificSuffix",
        "targetPath": "$.name.honorificSuffix",
        "optional": true
    },
    {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
    },
    {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
    },
    {
        "sourcePath": "$.emails[0].value",
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active"
    },
    {
        "sourcePath": "$.preferredLanguage",
        "targetPath": "$.preferredLanguage",

```

```

        "optional": true
      },
      {
        "sourcePath": "$.addresses",
        "targetPath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      },
      {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.timezone",
        "targetPath": "$.timezone",
        "optional": true
      },
      {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true
      }
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.members",
        "targetPath": "$.members",
        "preserveArrayWithSingleElement": true,
        "optional": true
      }
    ]
  }
}

```

5. Add a target system to provision users and groups to it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Business Network](#)

### 1.6.1.14 SAP Central Business Configuration

Follow this procedure to set up SAP Central Business Configuration (in short, CBC) as a source system.

## Prerequisites

You have created a technical user with administrator permissions that will be used to call the API of SAP Central Business Configuration for reading user and group information.

## Context

Create a CBC source system to read users, groups, and group members from it.

### i Note

Reading users is skipped by default. To enable it, go to the transformation and set the *"ignore": true* statement to **false**.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Central Business Configuration* as a source system. For more information, see [Add a System \[page 1477\]](#).



3. Choose the [Properties](#) tab to configure the connection settings for your system.

#### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your CBC system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Specify the technical user for your CBC system.
Password	Specify the password for your technical user.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.  For example: <b>https://mycbcaccount.authentication.us2.hana.ondemand.com/oauth/token</b>
(Optional) <code>cbc.user.filter</code>	When specified, only those CBC users matching the filter expression will be read.  Example: <b>name.familyName eq "Smith" and addresses.country eq "US"</b> <div><b>i</b> Note Using this property makes sense only if you have set the <a href="#">"ignore": true</a> statement to <b>false</b>.</div>
(Optional) <code>cbc.group.filter</code>	When specified, only those CBC groups matching the filter expression will be read.  Example: <b>displayName eq "ProjectTeam1" or "Employees2020"</b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Central Business Configuration](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your CBC system. For more information, see [Manage Transformations \[page 1494\]](#).

**Mapping logic** – the behavior of the default transformation logic is to read all user attributes from the source CBC system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

**Default transformation:**

#### Code Syntax

```
{
  "group": {
    "mappings": [
      {
        "sourcePath": "$.schema",
        "targetPath": "$.schema"
      },
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "'%cbc.group.prefix%' != 'null'",
            "function": "concatString",
            "prefix": "%cbc.group.prefix%"
          }
        ]
      }
    ],
    "sourcePath": "$.members",
    "preserveArrayWithSingleElement": true,
    "targetPath": "$.members"
  }
},
//To activate reading users, set "ignore":true statement to false.
"user": {
  "ignore": true,
  "mappings": [
    {
      "sourcePath": "$.schemas",
      "targetPath": "$.schemas"
    },
    {
      "sourcePath": "$.id",
      "targetVariable": "entityIdSourceSystem"
    },
    {
      "sourcePath": "$.userName",
      "targetPath": "$.userName",
      "correlationAttribute": true
    }
  ]
}
```

```

    "sourcePath": "$.name.givenName",
    "targetPath": "$.name.givenName"
  },
  {
    "sourcePath": "$.name.familyName",
    "targetPath": "$.name.familyName"
  },
  {
    "sourcePath": "$.active",
    "targetPath": "$.active"
  },
  {
    "sourcePath": "$.userType",
    "optional": true,
    "targetPath": "$.userType"
  }
]
}

```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Central Business Configuration – Collection](#) 

### 1.6.1.15 SAP Commerce Cloud

Follow this procedure to set up SAP Commerce Cloud as a source system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants

running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- In SAP Commerce Cloud, you have added an OAuth client with authorization grant **Client Credentials**. To learn how, see: [Configuring OAuth Client](#).

## Context

SAP Commerce Cloud is a highly flexible and modular platform for delivering modern customer experiences. It provides a standardized, automated, end-to-end solution that allows your projects to release code from repository to cloud.

You can use Identity Provisioning to configure SAP Commerce Cloud as a source system to read users and groups. These source systems consume SCIM 2.0 API provided by SAP Commerce Cloud. For more information, see [Commerce Cloud SCIM Web Services API Documentation](#).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Commerce Cloud* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	<p>Enter the URL to your SAP Commerce Cloud system.</p> <p>The URL follows the pattern:</p> <p><code>https://backoffice.&lt;tenant&gt;.model-t.cc.commerce.ondemand.com</code></p> <p>You can find the correct URL in the <a href="#">Environments</a> section of SAP Cloud Portal.</p>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the client ID to retrieve the OAuth access token for SAP Commerce Cloud.
Password	(Credential) Enter the client secret to retrieve the OAuth access token for SAP Commerce Cloud.

Property Name	Description & Value
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Commerce Cloud instance.
(Optional) <code>cc.user.filter</code>	<p>When specified, only those users matching the filter expression will be read. You can filter users by <b>userName</b>, <b>emails.value</b>, and <b>externalId</b>, according to the API syntax of SAP Commerce Cloud.</p> <p>For example:</p> <ul style="list-style-type: none"> <li><code>userName eq "johnbrown" and externalId eq "P000252"</code></li> <li><code>userName eq "johnbrown" and emails.value eq "johnbrown@email.com"</code></li> <li><code>userName eq "johnbrown" and emails.value eq "johnbrown@email.com" and externalId eq "P000252"</code></li> </ul> <div> <p><b>i Note</b></p> <p>These combinations are valid for both 'or' and 'and' operators.</p> </div>
(Optional) <code>cc.group.filter</code>	<p>When specified, only those groups matching the filter expression will be read.</p> <p>For example:</p> <p><code>displayName eq "ProjectTeam1" or "Students2018"</code></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Commerce Cloud* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Commerce Cloud system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Commerce Cloud SCIM Web Services API Documentation](#)

#### i Note

When configuring SAP Commerce Cloud as a source system, note that there may be mismatches between optional and required user attributes in the given target systems. For example, the email is an optional attribute in SAP Commerce Cloud but a required one in Identity Authentication. In such cases,

you need to review and adapt your source and/or target system transformations and configurations to reflect your needs.

## Note

Users that are created in SAP Commerce Cloud through the UI, don't have an ID and a username. As the ID is a mandatory attribute by SCIM specification, provisioning of those users to target systems will fail.

To solve this, create the users in SAP Commerce Cloud through API or implement ImpEx (import and export) for user ID provisioning, as described in [Configuring SAP Commerce Cloud](#).

### Default transformation:

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]"
      },
      {
        "sourcePath": "$.externalId",
        "optional": true,
        "targetPath": "$.externalId"
      },
      {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.nickName",
        "optional": true,
        "targetPath": "$.nickName"
      },
      {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$.name.middleName",
        "optional": true,
```

```

        "targetPath": "$.name.middleName"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.honorificSuffix",
        "optional": true,
        "targetPath": "$.name.honorificSuffix"
      },
      {
        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses",
        "functions": [
          {
            "type": "convertCountryCode",
            "outputFormat": "alpha2",
            "inputAttributes": [
              "country"
            ],
            "outputAttribute": "country"
          }
        ]
      },
      {
        "sourcePath": "$.userType",
        "optional": true,
        "targetPath": "$.userType"
      },
      {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails"
      },
      {
        "sourcePath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups"
      }
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName"
      }
    ]
  }
}

```

```

    "sourcePath": "$.members",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members"
  }
}
]
}

```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#).

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Cloud Identity Services Integration Architecture → cloudscimwebservices extension](#)

## 1.6.1.16 SAP Commissions

Follow this procedure to set up a source connector for SAP Commissions.

## Prerequisites

You have technical user credentials for an SAP Commissions system with read and write access permissions.

## Context

After fulfilling the prerequisites, follow the procedure below to add a source system for SAP Commissions to read users and user assignments to groups. This source system consumes SCIM 2.0 API provided by SAP Commissions.



## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Commissions* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the SAP Commissions SCIM API portal.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user for your SAP Commissions system.
Password	Enter the password for your SAP Commissions user.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

Type=*HTTP*

Authentication=*BasicAuthentication*

ProxyType=*Internet*

URL=*https://mycommissions.callidus.run/CallidusPortal*

User=*MyCommissionsUser*

Password=*\*\*\*\*\**

4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Commissions* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SCIM system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Commissions REST API](#) 

The behavior of the default transformation logic is to read all user attributes from the source SAP Commissions system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "targetVariable": "entityIdSourceSystem",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "optional": true
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "optional": true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "optional": true,
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "targetPath": "$.emails",
        "optional": true,
        "correlationAttribute": true
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath": "$.timezone",
      "optional": true,
      "targetPath": "$.timezone"
    },
    {
      "sourcePath": "$.addresses",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.addresses"
    },
    {
      "sourcePath": "$.groups",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.groups"
    },
    {
      "sourcePath": "$.schemas",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.schemas"
    },
    {
      "sourcePath": "$.locale",
      "optional": true,
      "targetPath": "$.locale",
      "functions": [
        {
          "type": "substring",
          "beginIndex": 0,
          "endIndex": 2
        }
      ]
    }
  ]
}

```

5. Now, add a target system to provision users and their group assignments to it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Commissions: Integration with SAP IdP](#)

## 1.6.1.17 SAP Concur

Follow this procedure to set up SAP Concur as a source system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

#### User Provisioning Service (UPS) v4 API with *Pre-2017 Authorization*

- You have created a technical user with administrator permissions that will be used to call the UPS v4 API for reading user account information. For more information, see [User Provisioning Service v4 API](#).
- You have registered a partner application in your SAP Concur system. You need the administrator permissions to register the application. For more information, see [Registering a Partner Application](#).

#### User Provisioning Service (UPS) v4 API or SAP Concur Identity v4 API with *OAuth 2.0* authentication

- You have an SAP Concur admin user with *Web Services Administrator* role assigned.
- Your SAP Concur admin user has obtained a *Company Request Token* and a *Company UUID* from the SAP Concur Company Request Token self-service tool.  
For more information, see [Configure an SAP Concur Entity as an IdP Target](#) → *Section 2: SAP Concur Company Request Token*.

### Context

Companies that use SAP Concur for managing and controlling travel expenses, invoices and other can use Identity Provisioning service to automate identity and access management for the SAP Concur solution. You can read identity data from SAP Concur and provision it to a target system of your choice (corporate user stores or SAP cloud user stores).

SAP Concur provides two APIs for its integration with Identity Provisioning: UPS v4 API and Identity v4 API (SCIM API). The value of `concur.api.version` property controls which API you use.

- When the value is set to **1**, or the property is not defined (typical for systems created before versioning was introduced on December 8, 2021), UPS v4 API is used. The UPS v4 API currently supports two authentication methods: *Pre-2017 Authorization* and *OAuth 2.0*. For more information on how to update to version 2, see: [Update Connector Version \[page 1484\]](#)
- When the value is set to **2**, Identity v4 API is used. This is the value that Identity Provisioning automatically sets for newly created systems after versioning was introduced on December 8, 2021. Identity v4 API supports provisioning of users with userUUID attribute which is generated by Identity Authentication at user creation.

To create SAP Concur as a source system, proceed as follows:

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Concur* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Version 1 Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter: <a href="https://www.concursolutions.com">https://www.concursolutions.com</a>
ProxyType	Enter: <i>Internet</i>
concur.api.version	Defines the version of SAP Concur API. Set the value to <i>1</i> to use UPS v4 API.
Authentication	Enter: <i>BasicAuthentication</i>  When using UPS v4 API (Version 1), two types of <i>BasicAuthentication</i> are supported: <ul style="list-style-type: none"><li>• <i>Pre-2017 Authorization</i> - Authentication based on Base-64 encoded Concur credentials (LoginID:Password) of the user. For more information, see <a href="#">Pre-2017 Authorization (Deprecated)</a>.</li><li>• <i>OAuth 2.0</i> - For more information, see <a href="#">Getting Started</a>.</li></ul>

Property Name	Description & Value
User	<p>Valid when <a href="#">Pre-2017 Authorization</a> is used.</p> <p>Enter the user ID of the SAP Concur technical user.</p>
Password	<p>Valid when <a href="#">Pre-2017 Authorization</a> is used.</p> <p>(Credential) Enter the password of the SAP Concur technical user.</p>
X-ConsumerKey	<p>Valid when <a href="#">Pre-2017 Authorization</a> is used.</p> <p>(Credential) Enter the key of the registered partner application (see the <b>Prerequisites</b> section).</p>
concur.datacenter	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Specify the SAP Concur data center your Identity Provisioning tenant belongs to. The following SAP Concur data centers are available:</p> <ul style="list-style-type: none"> <li>• us1</li> <li>• us2</li> <li>• eu1</li> <li>• eu2</li> <li>• emea</li> <li>• cn1</li> <li>• usg</li> <li>• int</li> </ul> <p>Based on the provided data center, Identity Provisioning configures the URL of the User Provisioning Service (UPS) v4 API or the SAP Concur Identity v4 API. For example, if you provide <b>us1</b>, the service will configure the URL in the following pattern: <code>us.api.concursolutions.com</code>.</p>

Property Name	Description & Value
<code>concur.authorization.code</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>(Credential) Enter the <a href="#">Company Request Token</a> and run a provisioning job within 24 hours from generating the token in the SAP Concur Company Request Token self-service tool. Otherwise, the token will expire, and you'll need a new one.</p> <p>After the first run of the job, Identity Provisioning fills in automatically a refresh token as the value of the <code>concur.refresh.token</code> property. If a provisioning job has not been run for six months, you'll again need to generate a new token.</p> <div> <p>→ Remember</p> <p>The company request token has a 24 hour validity. If this token expires, you must request a new token.</p> <p>The refresh token has a six month validity. Every time you run a provisioning job, the validity of the refresh token is extended with six months starting from the date of the last run. If you haven't run a provisioning job for six months, your refresh token will expire and you must request a new company request token.</p> </div>
<code>concur.company.id</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Enter the <a href="#">Company UUID</a> as described in the <i>Prerequisites</i> section.</p>
<code>concur.company.domain</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Enter your company domain.</p> <p>The username and the company domain are concatenated in the default transformation in the following format: <code>user@domain</code></p> <p>Your company domain is the part of your username behind the @ symbol. For example: <code>johnsmith@example.com</code></p>

#### Version 2 Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>

Property Name	Description & Value
<code>concur.datacenter</code>	<p>Specify the SAP Concur data center your Identity Provisioning tenant belongs to. The following SAP Concur data centers are available:</p> <ul style="list-style-type: none"> <li>• <code>us1</code></li> <li>• <code>us2</code></li> <li>• <code>eu1</code></li> <li>• <code>eu2</code></li> <li>• <code>emea</code></li> <li>• <code>cn1</code></li> <li>• <code>usg</code></li> <li>• <code>int</code></li> </ul> <p>Based on the provided data center, Identity Provisioning configures the URL of the User Provisioning Service (UPS) v4 API or the SAP Concur Identity v4 API. For example, if you provide <code>us1</code>, the service will configure the URL in the following pattern: <code>us.api.concursolutions.com</code>.</p>
<code>concur.api.version</code>	<p>Defines the version of SAP Concur API.</p> <p>Set the value to <code>2</code> to use Identity v4 API. This is the default value of the property.</p>
<code>concur.authorization.code</code>	<p>(Credential) Enter the <a href="#">Company Request Token</a> and run a provisioning job within 24 hours from generating the token in the SAP Concur Company Request Token self-service tool. Otherwise, the token will expire, and you'll need a new one.</p> <p>After the first run of the job, Identity Provisioning fills in automatically a refresh token as the value of the <code>concur.refresh.token</code> property. If a provisioning job has not been run for six months, you'll again need to generate a new token.</p> <div> <p>→ Remember</p> <p>The company request token has a 24 hour validity. If this token expires, you must request a new token.</p> <p>The refresh token has a six month validity. Every time you run a provisioning job, the validity of the refresh token is extended with six months starting from the date of the last run. If you haven't run a provisioning job for six months, your refresh token will expire and you must request a new company request token.</p> </div>



Property Name	Description & Value
<code>concur.company.id</code>	Enter the <a href="#">Company UUID</a> as described in the <i>Prerequisites</i> section.
<code>concur.company.domain</code>	<p>Enter your company domain.</p> <p>The username and the company domain are concatenated in the default transformation in the following format: <code>user@domain</code></p> <p>Your company domain is the part of your username behind the @ symbol. For example: <code>johnsmith@example.com</code></p>
(Optional) <code>concur.user.filter</code>	<p>When specified, only those users matching the filter expression will be read.</p> <p>For example:</p> <ul style="list-style-type: none"> <li><code>userName</code> eq "johnsmith@example.com" As the <code>userName</code> must be unique in SAP Concur, this filter returns only the user matching this <code>userName</code>.</li> <li><code>companyId</code> eq "aa067ada-71a9-4f57-8e98-9300b1c3171d" This filter returns all users in the company with this <code>companyId</code>.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Concur](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Concur source system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: Concur \(Users\)](#) 

**UPS v4 API:** [User Provisioning Service v4 API](#) 

**Identity v4 API:** [Identity v4](#) 

**Default transformation when using UPS v4 API (Version 1):**

≡ Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.EmployeeID",
```

```

        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem",
        "functions": [
            {
                "type": "compositeId",
                "subId": "$.LoginID"
            }
        ]
    },
    {
        "sourcePath": "$.EmployeeID",
        "targetPath": "$.userName",
        "correlationAttribute": true
    },
    {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
    },
    {
        "sourcePath": "$.PrimaryEmail",
        "targetPath": "$.emails[0].value",
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.FirstName",
        "optional": true,
        "targetPath": "$.name.givenName"
    },
    {
        "sourcePath": "$.LastName",
        "optional": true,
        "targetPath": "$.name.familyName"
    },
    {
        "sourcePath": "$.CellPhoneNumber",
        "optional": true,
        "targetPath": "$.phoneNumbers[0].value"
    },
    {
        "sourcePath": "$.LoginID",
        "correlationAttribute": true
    }
]
}
}

```

#### Default transformation when using Identity v4 API (Version 2):

##### Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.active",

```

```

        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails"
      },
      {
        "type": "remove",
        "targetPath": "$.emails[*].notifications"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "optional": true
      },
      {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.timezone",
        "optional": true,
        "targetPath": "$.timezone"
      },
      {
        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses"
      },
      {
        "sourcePath": "$.title",
        "optional": true,
        "targetPath": "$.title"
      },
      {
        "sourcePath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.phoneNumbers"
      },
      {
        "sourcePath": "$.emergencyContacts",
        "targetPath": "$.emergencyContacts",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath":
"$[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'companyId' ]",

```

```

        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['companyId']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['division']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']"
    }
]
}

```

```
}
```

5. Now, add a target system to provision users into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.18 SAP CPQ

Follow this procedure to set up SAP CPQ as a source system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

You have created a technical user with administrator permissions that will be used to call the API of SAP CPQ for reading user and group information.

## Context

Create an SAP CPQ source system to read users and groups from it.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

2. Add [SAP CPQ](#) as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the [Properties](#) tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your SAP CPQ system. It is the same as your SAP CPQ tenant URL. It must contain the domain name.  For example: <b>https://sample1234.mycpqdomain.com</b>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Specify the technical user for your SAP CPQ system. It must also contain the domain name, in format: <code>&lt;user_name&gt;#&lt;domain_name&gt;</code>  For example: <b>JohnSmith#MYCPQDOMAIN</b>
Password	(Credential) Specify the password for your technical user.
(Optional) <code>cpq.user.filter</code>	When specified, only those SAP CPQ users matching the filter expression will be read.  Example: <b>name.familyName eq "Smith" and addresses.country eq "US"</b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP CPQ](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

The behavior of the default transformation logic is to read all user attributes from the source SAP CPQ system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store

the unique ID of the identity. You can change the default transformation mapping rules to reflect your current setup of entities in your SAP CPQ system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP CPQ: SCIM API](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.meta",
        "targetPath": "$.meta"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails"
      },
      {
        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.addresses"
      },
      {
        "sourcePath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.phoneNumbers"
      },
      {
        "sourcePath": "$.groups",
```

```

        "preserveArrayWithSingleElement": true,
        "targetPath": "$.groups",
        "functions": [
            {
                "condition": "'%cpq.group.prefix%' !== 'null'",
                "function": "concatString",
                "applyOnElements": true,
                "applyOnAttribute": "display",
                "prefix": "%cpq.group.prefix%"
            }
        ]
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
    }
],
    "group": {
        "mappings": [
            {
                "sourcePath": "$.schemas",
                "targetPath": "$.schemas"
            },
            {
                "sourcePath": "$.meta",
                "targetPath": "$.meta"
            },
            {
                "sourcePath": "$.id",
                "targetPath": "$.id",
                "targetVariable": "entityIdSourceSystem"
            },
            {
                "sourcePath": "$.displayName",
                "targetPath": "$.displayName",
                "functions": [
                    {
                        "condition": "'%cpq.group.prefix%' !== 'null'",
                        "function": "concatString",
                        "prefix": "%cpq.group.prefix%"
                    }
                ]
            }
        ]
    },
    {
        "sourcePath": "$.members",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.members"
    }
]
}

```

- Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).



2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.19 SAP Data Custodian

Follow this procedure to set up SAP Data Custodian as a source system.

### Prerequisites


#### ! Restriction

This system is available for all standalone tenants and bundle tenants running on SAP Cloud Identity Services infrastructure. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- You have created an SAP Data Custodian tenant.
- You have created a user within your tenant with the required roles for your scenario.
- You have added your user to SAP Identity Service Management (SAP ISM).
- You have completed the Transparency and Control Service Onboarding Process or the Key Management Service Onboarding Process, depending on the scenario you want to implement. For more information, see [Transparency and Control Service Onboarding Process](#) and [Key Management Service Onboarding Process](#).

### Context

#### i Note

Currently, SAP Data Custodian connector is only available for selected customers who are approached by SAP. For more information, see [3319946](#) .

SAP Data Custodian is a robust Software as a Service (SaaS) solution that protects sensitive data stored in public, private, hybrid, and multicloud environments. This solution integrates with partnered public hyperscalers, SAP applications, and SAP managed clouds.

After fulfilling the prerequisites, follow the procedure below to add a source system for SAP Data Custodian to read users and user assignments to groups. This source system consumes SCIM 2.0 API provided by SAP Data Custodian.

### Procedure

1. Access the Identity Provisioning UI.

- [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Data Custodian* as a source system. For more information, see [Add a System \[page 1477\]](#).
  3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the SAP Data Custodian SCIM API portal.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth client key, created for your SAP Data Custodian tenant.
Password	Enter the OAuth client secret, created for your SAP Data Custodian tenant.
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Data Custodian instance, in format: <b>https://&lt;SAP_Data_Custodian_datacenter&gt;/api/v1/auth/token</b>
dc.group.filter	<p>This property filters groups by display name or externalId.</p> <p>When specified, only those groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>SAP_Data_Custodian_Auditor</i></li> <li>• <i>SAP_Data_Custodian_Service_Admin</i></li> <li>• <i>SAP_Data_Custodian_Key_Admin</i></li> <li>• <i>SAP_Data_Custodian_Key_User</i></li> </ul> <p>For example: <i>displayName eq "SAP_Data_Custodian_Auditor"</i></p>

Property Name	Value
<code>dc.user.filter</code>	<p>When specified, only those SAP Data Custodian users matching the filter expression will be read. You can filter users by <i>userName</i>, <i>displayName</i> or <i>externalId</i>.</p> <p><b>Possible values:</b></p> <p>For example: <i>userName eq "Smith.J"</i></p>
<code>dc.group.prefix</code>	<p>This property distinguishes SAP Data Custodian groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>DC_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Data Custodian source system and will be provisioned to the target system with the following name pattern: <b>DC_&lt;GroupDisplayName&gt;</b>. This way SAP Data Custodian groups in the target system will be distinguished from groups provisioned from other applications.</p> <p>If the property is not set, the SAP Data Custodian groups will be read and provisioned to the target system with their actual display names.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Data Custodian* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

##### **i** Note

During user creation, the expected format for *userName* is email. Hence, the primary email of the user is set for *userName*.

You can change the default transformation mapping rules to reflect your current setup of entities in your SCIM system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Data Custodian SCIM 2.0 API](#) 

**Default transformation:**

## Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "functions": [
          {
            "condition": "'%dc.group.prefix%' != 'null'",
            "function": "concatString",
            "applyOnElements": true,
            "applyOnAttribute": "display",
            "prefix": "%dc.group.prefix%"
          }
        ]
      }
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement": true
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath": "$.id",
      "targetVariable": "entityIdSourceSystem"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "functions": [
        {
          "condition": "'%dc.group.prefix%' != 'null'",
          "function": "concatString",
          "prefix": "%dc.group.prefix%"
        }
      ]
    },
    {
      "sourcePath": "$.members",
      "targetPath": "$.members",
      "optional": true,
      "preserveArrayWithSingleElement": true
    },
    {
      "sourcePath": "$.externalId",
      "targetPath": "$.externalId",
      "optional": true
    }
  ]
}

```

5. Now, add a target system to provision users and their group assignments to it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Data Custodian](#)

## 1.6.1.20 SAP Enterprise Portal

Follow this procedure to set up SAP Enterprise Portal as a source system.

### Prerequisites

- You are using SAP NetWeaver 7.5 Patch Level 22 and higher.
- You have prepared AS Java to use the SCIM application for REST API calls, as described in [Initial Setup](#). This means, you have activated the SCIM application and created and set user authorization.

### Context

SAP Enterprise Portal uses and builds on the user management of SAP NetWeaver AS for Java. Using SAP Enterprise Portal, organizations can give their employees, customers, partners, and suppliers a single point of access to the company applications, services, and information needed for conducting daily work. The portal offers business users the capability to easily create and manage portal pages and to generate their own portal content.

You can use Identity Provisioning to configure SAP Enterprise Portal as a source system where you can read users, groups and group assignments and provision them to target systems of your choice. In SAP Enterprise Portal, those entities correspond to users, Portal roles and user assignments to Portal roles, respectively.

#### i Note

When reading groups from SAP Enterprise Portal source system, only the exposed Portal roles are returned. When reading users, only user members of the exposed roles are returned.

In a typical scenario, SAP Enterprise Portal is configured as a source system and [SAP Build Work Zone, standard edition \[page 792\]](#) is configured as a target system. The SCIM Portal application for AS Java is developed to satisfy the SAP Enterprise Portal and SAP Build Work Zone, standard edition requirements for synchronization of exposed Portal roles and the user assignments to Portal roles.

For more information, see [SCIM Portal Application for AS Java](#)

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Enterprise Portal* as a source system. See: [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the SCIM API of your SAP Enterprise Portal system. It follows the pattern:  <code>http://&lt;host&gt;:&lt;port&gt;</code> of SAP NetWeaver where the SAP Enterprise Portal runs on.
ProxyType	Enter: <i>OnPremise</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user name of the technical user you have created following the <i>Create and Set User Authorization</i> step in the <i>Prerequisites</i> section.
Password	Enter the password of the technical user you have created following the <i>Create and Set User Authorization</i> step in the <i>Prerequisites</i> section.
ep.user.filter	When specified, only those SAP Enterprise Portal users matching the filter expression will be read. For more information, see <a href="#">Filtering</a> .
ep.group.filter	When specified, only those SAP Enterprise Portal groups matching the filter expression will be read. For more information, see <a href="#">Filtering</a> .

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map user and group attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Enterprise Portal* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SCIM Implementation for AS Java](#)

**Mapping logic** – the behavior of the default transformation logic is to read all user attributes from the source SAP Enterprise Portal system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

**Default transformation:**

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails",
        "optional": true
      },
      {
        "sourcePath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups"
      }
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "optional": true
      },
      {
        "sourcePath": "$.members",
        "targetPath": "$.members",
        "preserveArrayWithSingleElement": true,

```



```

    "optional": true
  },
  {
    "sourcePath": "$.schemas",
    "preserveArrayWithSingleElement": true,
    "targetPath": "$.schemas"
  }
]
}

```

5. Add a target system to provision users and groups to it. For example: [SAP Build Work Zone, standard edition \[page 792\]](#).

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Enterprise Portal](#)

### 1.6.1.21 SAP Fieldglass

Follow this procedure to set up a source connector for SAP Fieldglass.

## Prerequisites

You have created an API application key and a web service. To do that, follow the steps on page: [Create API Application Key or Web Service](#) and [Web Services Setup](#)

You will need the values of *Virtual Person Name (Username)* and *License Key* for the configuration of your source system (**step 3** below).

## Context

After fulfilling the prerequisites, follow the procedure below to add a source system for SAP Fieldglass to read users and groups from it. This source system consumes SCIM 2.0 API provided by SAP Fieldglass.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Fieldglass* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify your SAP Fieldglass environment URL. For example: <i>https://abc123.fgvms.com</i>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter your <i>Virtual Person Name (Username)</i> – see the <b>Prerequisites</b> section above.
Password	(Credential) Enter your <i>License Key</i> – see the <b>Prerequisites</b> section above.

Property Name	Value
OAuth2TokenServiceURL	<p>Enter your OAuth token URL in the following format:</p> <p><code>https://&lt;Environment_URL&gt;/api/oauth2/v2.0/token</code></p> <p>For example: <a href="https://abc123.fgvms.com/api/oauth2/v2.0/token">https://abc123.fgvms.com/api/oauth2/v2.0/token</a></p>
(Optional) fg.group.prefix	<p>This property distinguishes SAP Fieldglass groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>FG_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Fieldglass source system and will be provisioned to the target system with the following name pattern: <b>FG_&lt;GroupDisplayName&gt;</b>. This way SAP Fieldglass groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Fieldglass groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>FG_</b> prefix in their display name will be provisioned to SAP Fieldglass. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Fieldglass.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Fieldglass* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

The behavior of the default transformation logic is to read all user attributes from the source SAP Fieldglass system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

You can change the default transformation mapping rules to reflect your current setup of entities in your SCIM system. For more information, see:

[Manage Transformations \[page 1494\]](#)

**Default transformation:****Code Syntax**

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.title",
        "targetPath": "$.title",
        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.timezone",
        "optional": true,
        "targetPath": "$.timezone"
      },
      {
        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses"
      },
      {
        "sourcePath": "$.groups",

```

```

        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups",
        "functions": [
            {
                "condition": "%fg.group.prefix% != 'null'",
                "function": "concatString",
                "applyOnElements": true,
                "applyOnAttribute": "display",
                "prefix": "%fg.group.prefix%"
            }
        ]
    },
    {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
    },
    {
        "sourcePath": "$['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
    },
    {
        "sourcePath": "$['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true
    },
    {
        "sourcePath": "$['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
    },
    {
        "sourcePath": "$['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true
    },
    {
        "sourcePath": "$['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true
    },
    {
        "sourcePath": "$['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['value']",

```

```

        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "optional": true
    },
    {
        "sourcePath": "$['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
    }
]
},
// By default, the group mapping is inactive (ignored) but groups are
supported.
// To start provisioning groups, either delete the statement "ignore":
true, or set its value to false.
"group": {
    "ignore": true,
    "mappings": [
        {
            "sourcePath": "$.id",
            "targetPath": "$.id",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName",
            "functions": [
                {
                    "condition": "'%fg.group.prefix%' != 'null'",
                    "function": "concatString",
                    "prefix": "%fg.group.prefix%"
                }
            ]
        }
    ]
},
{
    "sourcePath": "$.members",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members"
},
{
    "sourcePath": "$.schemas",
    "preserveArrayWithSingleElement": true,
    "targetPath": "$.schemas"
}
]
}
}

```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.22 SAP Field Service Management

Follow this procedure to set up SAP Field Service Management as a source system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

You have OAuth credentials for SAP Field Service Management. For more information, see: [Generating Client ID & Secret](#)

### Context

SAP Field Service Management is a cloud-based solution that is used to resolve customers issues with end-to-end field service management. For example, it helps customers overcome resource limitations, such as having enough skilled technicians in all locations.

You can use the Identity Provisioning user interface (UI) to configure SAP Field Service Management as a source system where you can read users, groups, and group members.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Field Service Management* as a source system. See: [Add a System \[page 1477\]](#).

3. Choose the [Properties](#) tab to configure the connection settings for your system.

#### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your SAP Field Service Management system. It follows the pattern:  <code>https://&lt;cluster&gt;.coresystems.net</code>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client Id, created for your SAP Field Service Management system.
Password	(Credential) Enter the OAuth Client Secret, created for your SAP Field Service Management system.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.  For example: <code>https://&lt;fsm_account&gt;.coresuite.com/api/oauth2/v1/token</code>
(Optional) <code>fsm.user.filter</code>	When specified, only those SAP Field Service Management users matching the filter expression will be read.  Example: <code>name.familyName eq "Smith" and addresses.country eq "US"</code> <div><b>i</b> Note Using this property makes sense only if you have set the <code>"ignore": true</code> statement to <b>false</b>.</div>



Property Name	Value
(Optional) <code>fsm.group.prefix</code>	<p>This property distinguishes SAP Field Service Management groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>FSM_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Field Service Management source system and will be provisioned to the target system with the following name pattern: <b>FSM_&lt;GroupDisplayName&gt;</b>. This way SAP Field Service Management groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Field Service Management groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>FSM_</b> prefix in their display name will be provisioned to SAP Field Service Management. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Field Service Management.</li> </ul>
(Optional) <code>fsm.group.filter</code>	<p>When specified, only those SAP Field Service Management groups matching the filter expression will be read.</p> <p>Example: <b>displayName eq "ProjectTeam1" or "Employees2020"</b></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Field Service Management* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Field Service Management - SCIM API](#)

**Mapping logic** – the behavior of the default transformation logic is to read all user attributes from the source SAP Field Service Management system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails"
      },
      {
        "sourcePath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups",
        "functions": [
          {
            "condition": "'%fsm.group.prefix%' != 'null'",
            "function": "concatString",
            "applyOnElements": true,
            "applyOnAttribute": "display",
            "prefix": "%fsm.group.prefix%"
          }
        ]
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
```

```

    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "'%fsm.group.prefix%' != 'null'",
            "function": "concatString",
            "prefix": "%fsm.group.prefix%"
          }
        ]
      }
    ],
    "sourcePath": "$.members",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members"
  }
]
}

```

5. Add a target system to provision users and groups to it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Field Service Management – Collection](#) 

## 1.6.1.23 SAP Integrated Business Planning for Supply Chain

Follow this procedure to set up SAP Integrated Business Planning for Supply Chain (in short, SAP IBP) as a source system.

### Prerequisites

To establish the connection between Identity Provisioning and SAP Integrated Business Planning for Supply Chain, you need to set up the communication (user, system and arrangement) on SAP Integrated Business Planning for Supply Chain. You can do it now (as a prerequisite) or in the process of configuring SAP Integrated Business Planning for Supply Chain as a source system, as described in step 3.

### Context

SAP Integrated Business Planning for Supply Chain is a cloud-based solution that combines sales and operations planning (S&OP), forecasting and demand, response and supply, demand-driven replenishment, and inventory planning.

You can use Identity Provisioning to configure SAP IBP as a source system where you can read entities from and provision them to a target system. This scenario supports reading **business users** (employees), **user assignments**, and **business roles** (considered as *groups*).

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Integrated Business Planning for Supply Chain* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP Integrated Business Planning for Supply Chain and configure your authentication method (basic or certificate-based).

#### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Integrated Business Planning for Supply Chain source system, select the *Certificate* tab and choose ► *Generate* ► *Download* ►, as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP Integrated Business Planning for Supply Chain backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose *Scenario ID* SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

### **i** Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to your SAP IBP system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter your authentication method: <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>

Property Name	Description & Value
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div> <b>! Restriction</b>  Do not use special symbol ',' (comma) as it is not supported. </div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b>  Do not use special symbol ',' (comma) as it is not supported. </div>
ibp.skip.read.archived	<p>In the event of archived (disabled) entities in a source SAP IBP system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>
(Optional) ibp.roles.filter	<p>Enter OData filtering for reading roles in the SAP IBP system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ <b>4.5 Filter System Query Option</b></p>
(Optional) ibp.roles.page.size	<p>Indicate how many business roles (considered as <a href="#">groups</a>) per page to be read from your SAP IBP system.</p> <p>The value must be an integer number.</p>

Property Name	Description & Value
(Optional) <code>ibp.roles.prefix</code>	<p>This property distinguishes SAP Integrated Business Planning for Supply Chain roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <code>IBP_</code></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the SAP Integrated Business Planning for Supply Chain source system and will be provisioned to the target system with the following name pattern: <code>IBP_&lt;role_name&gt;</code>. This way SAP Integrated Business Planning for Supply Chain roles in the target system will be distinguished from roles provisioned from other applications.</p> <p>If the property is not set, the SAP Integrated Business Planning for Supply Chain roles will be read and provisioned to the target system with their actual role names.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://my1234567-api.scmibp.ondemand.com
User=MyIBPuser
Password=*****
ibp.date.variable.format=yyyy-MM-dd
ibp.skip.read.archived=true
ibp.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true
ibp.roles.page.size=30
```

## 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default

transformation for the [SAP IBP](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in your SAP IBP. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Integrated Business Planning API: Business User](#)

[SAP Business Accelerator Hub: SAP IBP](#) 

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.validityPeriod.startDate <= '${currentDate}') &&
($.validityPeriod.endDate > '${currentDate}')",
    "mappings": [
      {
        "sourcePath": "$.personID",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.personalInformation.firstName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.lastName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.personFullName",
        "targetPath": "$.name.formatted",
        "optional": true
      },
      {
        "sourcePath": "$.user.userName",
        "targetPath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "constant": true,
        "targetPath": "$.active"
      },
      {
        "condition": "$.user.lockedIndicator == 'true'",
        "constant": false,
        "targetPath": "$.active",
        "optional": true
      }
    ],
    // The following condition states that if a business user is outside its
    // validity period, it will be set as inactive.
    // That means, this user will not be able to log into the SAP IBP system.
  }
}
```



```

        "condition": "($.user.validityPeriod.startDate > '${currentDate}')
|| ('${currentDate}' > $.user.validityPeriod.endDate)",
        "constant": false,
        "optional": true,
        "targetPath": "$.active"
    },
    {
        "sourcePath": "$.workplaceInformation.emailAddress",
        "targetPath": "$.emails[0].value",
        "optional": true,
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.user.logonLanguageCode",
        "optional": true,
        "targetPath": "$.locale"
    },
    {
        "sourcePath": "$.PersonExternalID",
        "optional": true,
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.user.globalUserID",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
    },
    // The Identity Provisioning reads both users and user assignments from
    SAP IBP.
    {
        "sourcePath": "$.user.role",
        "optional": true,
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "functions": [
            {
                "condition": "'%ibp.roles.prefix%' !== 'null'",
                "function": "concatString",
                "applyOnElements": true,
                "prefix": "%ibp.roles.prefix%",
                "applyOnAttribute": "roleName",
                "assignToAttribute": "display"
            },
            {
                "condition": "'%ibp.roles.prefix%' === 'null'",
                "function": "concatString",
                "applyOnElements": true,
                "prefix": "",
                "applyOnAttribute": "roleName",
                "assignToAttribute": "display"
            },
            {
                "function": "concatString",
                "applyOnElements": true,
                "prefix": "",
                "applyOnAttribute": "roleName",
                "assignToAttribute": "value"
            }
        ]
    },
    {
        "type": "remove",
        "targetPath": "$.groups[*].roleName"
    },
    {
        "type": "valueMapping",
        "sourcePaths": [

```

```

    "$.user.timeZoneCode"
  ],
  "targetPath": "$.timezone",
  "defaultValue": "Europe/Berlin",
  "valueMappings": [
    {
      "key": [
        "WDFI"
      ],
      "mappedValue": "Europe/Berlin"
    },
    {
      "key": [
        "ISRAEL"
      ],
      "mappedValue": "Asia/Jerusalem"
    },
    {
      "key": [
        "RUS03"
      ],
      "mappedValue": "Europe/Moscow"
    },
    {
      "key": [
        "AUSNSW"
      ],
      "mappedValue": "Australia/Sydney"
    },
    {
      "key": [
        "UTC+4"
      ],
      "mappedValue": "Asia/Dubai"
    },
    {
      "key": [
        "BRAZIL"
      ],
      "mappedValue": "America/Sao_Paulo"
    },
    {
      "key": [
        "BRZLEA"
      ],
      "mappedValue": "America/Sao_Paulo"
    },
    {
      "key": [
        "MSTNO"
      ],
      "mappedValue": "America/Phoenix"
    },
    {
      "key": [
        "EST"
      ],
      "mappedValue": "America/New_York"
    },
    {
      "key": [
        "UTC"
      ],
      "mappedValue": "Etc/UTC"
    },
    {
      "key": [
        "UTC+3"
      ]
    }
  ]
}

```

```

        ],
        "mappedValue": "Asia/Riyadh"
    },
    {
        "key": [
            "EST_"
        ],
        "mappedValue": "America/Toronto"
    },
    {
        "key": [
            "UTC+8"
        ],
        "mappedValue": "Asia/Shanghai"
    },
    {
        "key": [
            "JAPAN"
        ],
        "mappedValue": "Asia/Tokyo"
    }
    ]
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.businessPartnerRoleCode"
    ],
    "targetPath": "$.userType",
    "defaultValue": "Employee",
    "valueMappings": [
        {
            "key": [
                "BUP003"
            ],
            "mappedValue": "Employee"
        }
    ]
}
],
},
"group": {
    "mappings": [
        {
            "sourcePath": "$.ID",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "sourcePath": "$.ID",
            "functions": [
                {
                    "condition": "'%ibp.roles.prefix%' != 'null'",
                    "function": "concatString",
                    "prefix": "%ibp.roles.prefix%"
                }
            ],
            "targetPath": "$.displayName"
        },
        {
            "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath": "$.schemas[0]"
        },
        {
            "sourcePath": "$.to_BusinessUserAssignment.results",
            "optional": true,
            "preserveArrayWithSingleElement": true,
            "targetPath": "$.members"
        }
    ],
}

```

```

    {
      "type": "remove",
      "targetPath": "$.members[*].__metadata"
    },
    {
      "type": "rename",
      "constant": "value",
      "targetPath": "$.members[*].PersonID"
    },
    {
      "constant": "User",
      "targetPath": "$.members[*].type"
    }
  ]
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource. To learn how, see [Guided Answers: Business Role Description](#).

- Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Integrated Business Planning](#)

### 1.6.1.24 SAP Jam Collaboration

Follow this procedure to set up SAP Jam Collaboration as a source system.

## Prerequisites

You get OAuth credentials for SAP Jam Collaboration. If your SAP Jam tenant is of "SCIM provisioning" type, an OAuth client is automatically created for it, with the name **SCIM API Client**. To find this client:

1. Go to the SAP Jam Collaboration admin panel.
2. Choose **Integrations** > **OAuth Clients**.
3. For **SCIM API Client**, choose **View**.
4. Save the **Key** and **Secret** values – you'll need them later while configuring your SAP Jam Collaboration provisioning system.

To learn more, see: [SAP Jam: Add an OAuth Client](#)

## Context

After fulfilling the prerequisites, follow the procedure below to create a source SAP Jam Collaboration system to read users and groups.

These source systems consume SCIM 2.0 API provided by SAP Jam Collaboration.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add **SAP Jam Collaboration** as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the **Properties** tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a **connectivity destination** in your subaccount in the SAP BTP cockpit, and then select it from the **Destination Name** combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the **Properties** tab, the value set in the **Properties** tab is considered with higher priority.

We recommend that you use the **Properties** tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <b>HTTP</b>
URL	Enter the URL related to your SAP Jam system, in format: <b>https://&lt;SAP_Jam_datacenter&gt;.sapjam.com</b>  For example: <b>https://jam4.sapjam.com</b>
ProxyType	Enter: <b>Internet</b>

Property Name	Description & Value
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth client key, created for your SAP Jam tenant (see <b>Prerequisites</b> ).
Password	Enter the OAuth client secret, created for your SAP Jam tenant (see <b>Prerequisites</b> ).
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Jam instance, in format: <b>https://&lt;SAP_Jam_datacenter&gt;/api/v1/auth/token</b>  For example: <i>https://jam4.sapjam.com/api/v1/auth/token</i>
(Optional) jam.group.prefix	<p>This property distinguishes SAP Jam Collaboration groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SJC_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Jam Collaboration source system and will be provisioned to the target system with the following name pattern: <b>SJC_&lt;GroupDisplayName&gt;</b>. This way SAP Jam Collaboration groups in the target system will be distinguished from groups provisioned from other applications.</p> <p>If the property is not set, the SAP Jam Collaboration groups will be read and provisioned to the target system with their actual display names.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Jam Collaboration* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Jam Collaboration system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: SAP Jam Collaboration](#) 

**Default transformation:**

## Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "targetPath": "$",
        "sourcePath": "$"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "type": "remove",
        "targetPath": "$.id"
      },
      {
        "type": "remove",
        "targetPath": "$.meta"
      }
    ]
  },
  "group": {
    "ignore": true,
    "mappings": [
      {
        "targetPath": "$",
        "sourcePath": "$"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "'%jam.group.prefix%' != 'null'",
            "function": "concatString",
            "prefix": "%jam.group.prefix%"
          }
        ]
      },
      {
        "type": "remove",
        "targetPath": "$.id"
      },
      {
        "type": "remove",
        "targetPath": "$.meta"
      }
    ]
  }
}
```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe to the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during your jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.25 SAP Market Communication for Utilities

Follow this procedure to set up SAP Market Communication for Utilities as a source system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

To establish the connection between Identity Provisioning and SAP Market Communication for Utilities, you need to set up the communication user in SAP BTP ABAP environment. You can do it now (as a prerequisite) or in the process of configuring SAP Market Communication for Utilities as a source system, as described in step 3.

## Context

The SAP Market Communication for Utilities application is based on SAP BTP ABAP environment. You can use Identity Provisioning to configure SAP Market Communication for Utilities as a source system to read entities from and provision them to a target system.

This scenario supports reading **business users** (Employee), **user assignments**, and **business roles** (which are considered as *groups*).




## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Market Communication for Utilities* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP Market Communication for Utilities and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Market Communication for Utilities source system, select the *Certificate* tab and choose **Generate** > **Download** , as described in [Generate and Manage Certificates for Outbound Connection \[page 1507\]](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP BTP ABAP environment backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose *Scenario ID* SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

For more information, see [Maintain a Communication Arrangement for Inbound Communication](#) .

### i Note

The communication scenario *SAP\_COM\_0193* is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the *Properties* tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the API URL to your SAP Market Communication for Utilities system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter your authentication method: <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>

Property Name	Description & Value
<code>maco.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source SAP Market Communication for Utilities system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
<code>ips.date.variable.format</code>	<code>yyyy-MM-dd</code>
(Optional) <code>maco.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP Market Communication for Utilities system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ <b>4.5 Filter System Query Option</b></p>
(Optional) <code>maco.roles.page.size</code>	<p>Indicate how many business roles (considered as <i>groups</i>) per page to be read from your SAP Market Communication for Utilities system.</p> <p>The value must be an integer number.</p>
(Optional) <code>maco.roles.prefix</code>	<p>This property distinguishes SAP Market Communication for Utilities roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SMC_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the SAP Market Communication for Utilities source system and will be provisioned to the target system with the following name pattern: <b>SMC_&lt;role_name&gt;</b>. This way SAP Market Communication for Utilities roles in the target system will be distinguished from roles provisioned from other applications.</p> <p>If the property is not set, the SAP Market Communication for Utilities roles will be read and provisioned to the target system with their actual role names.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP

Authentication=BasicAuthentication

ProxyType=Internet

URL=https://12345-aaaaa-3333.abap.hana.ondemand.com

User=MyMaCoUser

Password=*****

ips.date.variable.format=yyyy-MM-dd

maco.skip.read.archived=true

maco.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true

maco.roles.page.size=30
```

---

##### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Market Communication for Utilities](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in your SAP Market Communication for Utilities. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

##### Default transformation:

###### Code Syntax

```
{
  "user": {
    "condition": "($.validityPeriod.startDate <= '${currentDate}') &&
($.validityPeriod.endDate > '${currentDate}')",
    "mappings": [
      {
        "sourcePath": "$.personID",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.personalInformation.firstName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.lastName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.middleName",
        "targetPath": "$.name.middleName",
```

```

    "optional": true
  },
  {
    "sourcePath": "$.personalInformation.personFullName",
    "targetPath": $.name.formatted,
    "optional": true
  },
  {
    "sourcePath": $.user.userName,
    "targetPath": $.userName,
    "optional": true,
    "correlationAttribute": true
  },
  {
    "constant": true,
    "targetPath": $.active
  },
  {
    "condition": $.user.lockedIndicator == 'true',
    "constant": false,
    "targetPath": $.active,
    "optional": true
  },
  {
    "condition": "($.user.validityPeriod.startDate > '${currentDate}')
|| ('${currentDate}' > $.user.validityPeriod.endDate)",
    "constant": false,
    "optional": true,
    "targetPath": $.active
  },
  {
    "sourcePath": $.workplaceInformation.emailAddress,
    "targetPath": $.emails[0].value,
    "optional": true,
    "correlationAttribute": true
  },
  {
    "sourcePath": $.user.logonLanguageCode,
    "optional": true,
    "targetPath": $.locale
  },
  {
    "sourcePath": $.PersonExternalID,
    "optional": true,
    "correlationAttribute": true
  },
  {
    "sourcePath": $.user.role,
    "optional": true,
    "targetPath": $.groups,
    "preserveArrayWithSingleElement": true,
    "functions": [
      {
        "condition": "'%maco.roles.prefix%' != 'null'",
        "function": "concatString",
        "applyOnElements": true,
        "prefix": "%maco.roles.prefix%",
        "applyOnAttribute": "roleName",
        "assignToAttribute": "display"
      },
      {
        "condition": "'%maco.roles.prefix%' == 'null'",
        "function": "concatString",
        "applyOnElements": true,
        "prefix": "",
        "applyOnAttribute": "roleName",
        "assignToAttribute": "display"
      }
    ]
  },

```

```

        {
            "function": "concatString",
            "applyOnElements": true,
            "prefix": "",
            "applyOnAttribute": "roleName",
            "assignToAttribute": "value"
        }
    ]
},
{
    "type": "remove",
    "targetPath": "$.groups[*].roleName"
},
{
    "sourcePath": "$.user.globalUserID",
    "optional": true,
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.user.timeZoneCode"
    ],
    "targetPath": "$.timezone",
    "defaultValue": "Europe/Berlin",
    "valueMappings": [
        {
            "key": [
                "WDF"
            ],
            "mappedValue": "Europe/Berlin"
        },
        {
            "key": [
                "ISRAEL"
            ],
            "mappedValue": "Asia/Jerusalem"
        },
        {
            "key": [
                "RUS03"
            ],
            "mappedValue": "Europe/Moscow"
        },
        {
            "key": [
                "AUSNSW"
            ],
            "mappedValue": "Australia/Sydney"
        },
        {
            "key": [
                "UTC+4"
            ],
            "mappedValue": "Asia/Dubai"
        },
        {
            "key": [
                "BRAZIL"
            ],
            "mappedValue": "America/Sao_Paulo"
        },
        {
            "key": [
                "BRZLEA"
            ],
            "mappedValue": "America/Sao_Paulo"
        }
    ]
}

```

```

    },
    {
      "key": [
        "MSTNO"
      ],
      "mappedValue": "America/Phoenix"
    },
    {
      "key": [
        "EST"
      ],
      "mappedValue": "America/New_York"
    },
    {
      "key": [
        "UTC"
      ],
      "mappedValue": "Etc/UTC"
    },
    {
      "key": [
        "UTC+3"
      ],
      "mappedValue": "Asia/Riyadh"
    },
    {
      "key": [
        "EST_"
      ],
      "mappedValue": "America/Toronto"
    },
    {
      "key": [
        "UTC+8"
      ],
      "mappedValue": "Asia/Shanghai"
    },
    {
      "key": [
        "JAPAN"
      ],
      "mappedValue": "Asia/Tokyo"
    }
  ]
},
{
  "type": "valueMapping",
  "sourcePaths": [
    "$.businessPartnerRoleCode"
  ],
  "targetPath": "$.userType",
  "defaultValue": "Employee",
  "valueMappings": [
    {
      "key": [
        "BUP003"
      ],
      "mappedValue": "Employee"
    }
  ]
}
]
},
"group": {
  "mappings": [
    {
      "sourcePath": "$.ID",
      "targetVariable": "entityIdSourceSystem"
    }
  ]
}
}

```

```

    },
    {
      "sourcePath": "$.ID",
      "functions": [
        {
          "condition": "'%maco.roles.prefix%' !== 'null'",
          "function": "concatString",
          "prefix": "%maco.roles.prefix%"
        }
      ],
      "targetPath": "$.displayName"
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.to_BusinessUserAssignment.results",
      "optional": true,
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.members"
    },
    {
      "type": "remove",
      "targetPath": "$.members[*].__metadata"
    },
    {
      "type": "rename",
      "constant": "value",
      "targetPath": "$.members[*].PersonID"
    },
    {
      "constant": "User",
      "targetPath": "$.members[*].type"
    }
  ]
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource. To learn how, see [Guided Answers: Business Role Description](#).

- Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).



## Related Information

[SAP S/4HANA Cloud Documentation](#)

### 1.6.1.26 SAP Marketing Cloud

Follow this procedure to set up SAP Marketing Cloud as a source system.

## Prerequisites

To establish the connection between Identity Provisioning and SAP Marketing Cloud, you need to set up the communication (user, system and arrangement) on SAP Marketing Cloud. You can do it now (as a prerequisite) or in the process of configuring SAP Marketing Cloud as a source system, as described in step 3.

## Context




You can use SAP Marketing Cloud to read entities from it and provision them to a target system. This scenario supports reading **users**, **user assignments**, and **Business roles** (which are considered as *groups*).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Marketing Cloud* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP Marketing Cloud and configure your authentication method (basic or certificate-based).

#### Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Marketing Cloud source system, select the *Certificate* tab and choose  *Generate*  *Download* , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP Marketing Cloud backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose *Scenario ID* SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

### i Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to your SAP Marketing Cloud system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter your authentication method: <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>

Property Name	Description & Value
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div> <b>! Restriction</b> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
marketing.cloud.skip.read.archived	<p>In the event of archived (disabled) entities in a source SAP Marketing Cloud system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>
(Optional)marketing.cloud.roles.filter	<p>Enter OData filtering for reading roles in the SAP Marketing Cloud system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ <b>4.5 Filter System Query Option</b></p>
(Optional)marketing.cloud.roles.page.size	<p>Indicate how many business roles (considered as <a href="#">groups</a>) per page to be read from your SAP Marketing Cloud system.</p> <p>The value must be an integer number.</p>

Property Name	Description & Value
(Optional)marketing.cloud.roles.prefix	<p>This property distinguishes SAP Marketing Cloud roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SMKC_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the source system</b>, the prefix will be prepended to the name of the roles that are read from the SAP Marketing Cloud source system and will be provisioned to the target system with the following name pattern: <b>SMKC_&lt;role_name&gt;</b> . This way SAP Marketing Cloud roles in the target system will be distinguished from roles provisioned from other applications.</p> <p>If the property is not set, the SAP Marketing Cloud roles will be read and provisioned to the target system with their actual role names.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://my1234567-api.s4hana.ondemand.com
User=MyMarketingCloudUser
Password=*****
ips.date.variable.format=yyyy-MM-dd
marketing.skip.read.archived=true
marketing.cloud.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true
marketing.cloud.roles.page.size=30
```

#### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Marketing Cloud* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in your SAP Marketing Cloud. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Marketing Cloud API: Business User](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.validityPeriod.startDate <= '${currentDate}') &&
($.validityPeriod.endDate > '${currentDate}')",
    "mappings": [
      {
        "sourcePath": "$.personID",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.personalInformation.firstName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.lastName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.personFullName",
        "targetPath": "$.name.formatted",
        "optional": true
      },
      {
        "sourcePath": "$.user.userName",
        "targetPath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "constant": true,
        "targetPath": "$.active"
      },
      {
        "condition": "$.user.lockedIndicator == 'true'",
        "constant": false,
        "targetPath": "$.active",
        "optional": true
      }
    ],
    // The following condition states that if a business user is outside its
    // validity period, it will be set as inactive.
    // That means, this user will not be able to log into the SAP Marketing
    // Cloud system.
    {
      "condition": "($.user.validityPeriod.startDate > '${currentDate}')
|| ('${currentDate}' > $.user.validityPeriod.endDate)",
      "constant": false,
      "optional": true,
      "targetPath": "$.active"
    }
  }
}
```

```

    },
    {
      "sourcePath": "$.workplaceInformation.emailAddress",
      "targetPath": "$.emails[0].value",
      "optional": true,
      "correlationAttribute": true
    },
    {
      "sourcePath": $.user.logonLanguageCode,
      "optional": true,
      "targetPath": $.locale
    },
    {
      "sourcePath": $.PersonExternalID,
      "optional": true,
      "correlationAttribute": true
    },
    {
      "sourcePath": $.user.globalUserID,
      "optional": true,
      "targetPath": $
    }
  ],
  ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']
},
// The Identity Provisioning reads both users and user assignments from
SAP Marketing Cloud.
{
  "sourcePath": $.user.role,
  "optional": true,
  "targetPath": $.groups,
  "preserveArrayWithSingleElement": true,
  "functions": [
    {
      "condition": '%"marketing.cloud.roles.prefix%' != 'null',
      "function": "concatString",
      "applyOnElements": true,
      "prefix": '%"marketing.cloud.roles.prefix%',
      "applyOnAttribute": "roleName",
      "assignToAttribute": "display"
    },
    {
      "condition": '%"marketing.cloud.roles.prefix%' == 'null',
      "function": "concatString",
      "applyOnElements": true,
      "prefix": "",
      "applyOnAttribute": "roleName",
      "assignToAttribute": "display"
    },
    {
      "function": "concatString",
      "applyOnElements": true,
      "prefix": "",
      "applyOnAttribute": "roleName",
      "assignToAttribute": "value"
    }
  ]
},
{
  "type": "remove",
  "targetPath": $.groups[*].roleName
},
{
  "type": "valueMapping",
  "sourcePaths": [
    $.user.timeZoneCode
  ],
  "targetPath": $.timezone,
  "defaultValue": "Europe/Berlin",
  "valueMappings": [

```

```

{
  "key": [
    "WDFT"
  ],
  "mappedValue": "Europe/Berlin"
},
{
  "key": [
    "ISRAEL"
  ],
  "mappedValue": "Asia/Jerusalem"
},
{
  "key": [
    "RUS03"
  ],
  "mappedValue": "Europe/Moscow"
},
{
  "key": [
    "AUSNSW"
  ],
  "mappedValue": "Australia/Sydney"
},
{
  "key": [
    "UTC+4"
  ],
  "mappedValue": "Asia/Dubai"
},
{
  "key": [
    "BRAZIL"
  ],
  "mappedValue": "America/Sao_Paulo"
},
{
  "key": [
    "BRZLEA"
  ],
  "mappedValue": "America/Sao_Paulo"
},
{
  "key": [
    "MSTNO"
  ],
  "mappedValue": "America/Phoenix"
},
{
  "key": [
    "EST"
  ],
  "mappedValue": "America/New_York"
},
{
  "key": [
    "UTC"
  ],
  "mappedValue": "Etc/UTC"
},
{
  "key": [
    "UTC+3"
  ],
  "mappedValue": "Asia/Riyadh"
},
{
  "key": [

```

```

        "EST_"
      ],
      "mappedValue": "America/Toronto"
    },
    {
      "key": [
        "UTC+8"
      ],
      "mappedValue": "Asia/Shanghai"
    },
    {
      "key": [
        "JAPAN"
      ],
      "mappedValue": "Asia/Tokyo"
    }
  ]
},
{
  "type": "valueMapping",
  "sourcePaths": [
    "$.businessPartnerRoleCode"
  ],
  "targetPath": "$.userType",
  "defaultValue": "Employee",
  "valueMappings": [
    {
      "key": [
        "BUP003"
      ],
      "mappedValue": "Employee"
    },
    {
      "key": [
        "BBP005"
      ],
      "mappedValue": "Contingent Worker"
    }
  ]
}
],
},
"group": {
  "mappings": [
    {
      "sourcePath": "$.ID",
      "targetVariable": "entityIdSourceSystem"
    },
    {
      "sourcePath": "$.ID",
      "functions": [
        {
          "condition": "'%marketing.cloud.roles.prefix%' != 'null'",
          "function": "concatString",
          "prefix": "%marketing.cloud.roles.prefix%"
        }
      ],
      "targetPath": "$.displayName"
    }
  ],
  "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
  "targetPath": "$.schemas[0]"
},
{
  "sourcePath": "$.to_BusinessUserAssignment.results",
  "optional": true,
  "preserveArrayWithSingleElement": true,
  "targetPath": "$.members"
}

```



```

    },
    {
      "type": "remove",
      "targetPath": "$.members[*].__metadata"
    },
    {
      "type": "rename",
      "constant": "value",
      "targetPath": "$.members[*].PersonID"
    },
    {
      "constant": "User",
      "targetPath": "$.members[*].type"
    }
  ]
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource. To learn how, see [Guided Answers: Business Role Description](#).

- Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA Cloud Documentation](#)

## 1.6.1.27 SAP Master Data Integration

Follow this procedure to set up SAP Master Data Integration (in short, MDI) as a source system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants

running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- You have created a tenant in a subaccount on SAP BTP, Cloud Foundry environment. You can create your own tenant (free of charge), or integrate with an existing one.
- You have created a service instance for MDI in the subaccount in order to connect a new system to your tenant and read user account information from it. To learn how, see: [Creating Service Instances](#)
- You have created a service key in this instance, which contains the necessary credentials to connect to the MDI service. Creating multiple service keys in the same service instance is not supported. To learn how, see: [Creating service Instances](#)

#### → Tip

The `serviceKey` payload provides you with the following properties that you will later need for your system configuration:

- `uri` = URL
- `uaa.url` = OAuth2TokenServiceURL
- `uaa.clientid` = User
- `clientsecret` = Password

## Context

As part of SAP's data model and integration unification strategy, SAP BTP Integration Suite has [Master Data Integration](#) to enable a harmonized integration and distribution of different master data objects and data between SAP solutions. This includes master data for business partners, cost centers, and workforce data. Workforce data is provided for integration scenarios that need data from SAP SuccessFactors Employee Central or other HR systems.

To learn more, see: [Integrating SAP SuccessFactors Employee Central with SAP Master Data Integration](#)

You can read users from MDI and provision it to a target system of your choice.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add [SAP Master Data Integration](#) as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the [Properties](#) tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Enter the URL to the relevant integration application running in the relevant region of SAP Business Technology Platform.</p> <p>See the <b>Prerequisites</b> section → <a href="#">serviceKey</a> tip.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	<p>Enter the technical user that has access to the API of your MDI service.</p> <p>See the <b>Prerequisites</b> section → <a href="#">serviceKey</a> tip.</p>
Password	<p>Enter the password for this technical user.</p> <p>See the <b>Prerequisites</b> section → <a href="#">serviceKey</a> tip.</p>
OAuth2TokenServiceURL	<p>Enter the OAuth 2.0 Token Service URL.</p> <p>See the <b>Prerequisites</b> section → <a href="#">serviceKey</a> tip.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Master Data Integration](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your MDI source system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Field Mapping Between Employee Central and SAP Master Data Integration](#)

## Default transformation:

### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem",
        "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User' ][ 'personGUID' ]",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.externalId",
        "targetPath":
"$[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
        "optional": true
      },
      {
        "sourcePath":
"$$.profileDetail[0].content.scriptedProfileDetails[0].firstName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$$.profileDetail[0].content.scriptedProfileDetails[0].lastName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath":
"$$.profileDetail[0].content.scriptedProfileDetails[0].middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.userAccount.userName",
        "targetPath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "constant": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails[?(@.isDefault == true)].address",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails"
      },
      {
        "targetPath": "$.emails[*].usage",
        "type": "remove"
      },
      {
        "targetPath": "$.emails[*].address",
```

```

        "type": "rename",
        "constant": "value"
      },
      {
        "targetPath": "$.emails[*].isDefault",
        "type": "rename",
        "constant": "primary"
      },
      {
        "constant": [
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User",
          "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
        ],
        "targetPath": "$.schemas"
      }
    ]
  }
}

```

- Now, add a target system to provision users into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Community: The New Master Data Integration Service for SAP SuccessFactors](#) 

### 1.6.1.28 SAP S/4HANA Cloud

Follow this procedure to set up SAP S/4HANA Cloud as a source system.

## Prerequisites

To establish the connection between Identity Provisioning and SAP S/4HANA Cloud, you need to set up the communication (user, system and arrangement) on SAP S/4HANA Cloud. You can do it now (as a prerequisite) or in the process of configuring SAP S/4HANA Cloud as a source system, as described in step 3.

## Context

SAP S/4HANA Cloud is a complete enterprise resource planning (ERP) system with built-in intelligent technologies and advanced analytics.

You can use Identity Provisioning to configure SAP S/4HANA Cloud as a source system where you can read entities from and provision them to target systems of your choice. This scenario supports reading **business users** (Employee, Contingent Worker), **user assignments**, and **business roles** (which are considered as *groups*).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP S/4HANA Cloud* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP S/4HANA Cloud and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP S/4HANA source system, select the *Certificate* tab and choose ► *Generate* ► *Download* ►, as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP S/4HANA backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose *Scenario ID* SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

### i Note

The communication scenario *SAP\_COM\_0193* is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

#### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Enter the SAP S/4HANA Cloud API URL.</p> <p>You can find the correct URL in the <a href="#">API-URL</a> field of the communication arrangement set up for communication scenario SAP_COM_0193.</p> <p>For example: <code>https://my123456-api.s4hana.ondemand.com</code></p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>

Property Name	Description & Value
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b>  Do not use special symbol ',' (comma) as it is not supported. </div>
s4hana.cloud.skip.read.archived	<p>In the event of archived (disabled) entities in a source SAP S/4HANA Cloud system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>
s4hana.cloud.api.version	<p>The version of the system API you use.</p> <p>Version <a href="#">1</a> means your SAP S/4HANA Cloud system uses <a href="#">SAP_COM_0193</a> communication arrangement.</p>
(Optional) s4hana.cloud.roles.filter	<p>Enter OData filtering for reading roles in the S/4HANA system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ <a href="#">4.5 Filter System Query Option</a></p>
(Optional) s4hana.cloud.roles.page.size	<p>Indicate how many business roles (considered as <a href="#">groups</a>) per page to be read from your SAP S/4HANA Cloud system.</p> <p>The value must be an integer number.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

Exemplary destination:



```
Type=HTTP

Authentication=BasicAuthentication

ProxyType=Internet

URL=https://my1234567-api.s4hana.ondemand.com

User=MyS4HANAUser

Password=*****

s4hana.cloud.api.version=1

ips.date.variable.format=yyyy-MM-dd

s4hana.cloud.skip.read.archived=true

s4hana.cloud.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true

s4hana.cloud.roles.page.size=30
```

---

5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP S/4HANA Cloud](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in your SAP S/4HANA Cloud. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

**Default transformation:**

**Code Syntax**

```
{
  "user": {
    "condition": "($.user.validityPeriod.startDate <= '${currentDate}') &&
 ($.user.validityPeriod.endDate > '${currentDate}')",
    "mappings": [
      {
        "sourcePath": "$.personID",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.personalInformation.firstName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.lastName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.middleName",
        "targetPath": "$.name.middleName",
```

```

        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.personFullName",
        "targetPath": $.name.formatted,
        "optional": true
      },
      {
        "sourcePath": $.user.userName,
        "targetPath": $.userName,
        "optional": true,
        "correlationAttribute": true
      },
      {
        "constant": true,
        "targetPath": $.active
      },
      {
        "condition": $.user.lockedIndicator == 'true',
        "constant": false,
        "targetPath": $.active,
        "optional": true
      },
      {
        "sourcePath": $.workplaceInformation.emailAddress,
        "targetPath": $.emails[0].value,
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": $.user.logonLanguageCode,
        "optional": true,
        "targetPath": $.locale
      },
      {
        "sourcePath": $.PersonExternalID,
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": $.user.globalUserID,
        "optional": true,
        "targetPath": $
      },
      [ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]
    ],
    {
      "sourcePath": $.user.role,
      "targetPath": $.groups,
      "preserveArrayWithSingleElement": true,
      "functions": [
        {
          "condition": "'%s4hana.cloud.roles.prefix%' != 'null'",
          "function": "concatString",
          "applyOnElements": true,
          "prefix": "%s4hana.cloud.roles.prefix%",
          "applyOnAttribute": "roleName",
          "assignToAttribute": "display"
        },
        {
          "condition": "'%s4hana.cloud.roles.prefix%' == 'null'",
          "function": "concatString",
          "applyOnElements": true,
          "prefix": "",
          "applyOnAttribute": "roleName",
          "assignToAttribute": "display"
        }
      ],
      {
        "function": "concatString",

```

```

        "applyOnElements": true,
        "prefix": "",
        "applyOnAttribute": "roleName",
        "assignToAttribute": "value"
    }
}
],
},
{
    "type": "remove",
    "targetPath": "$.groups[*].roleName"
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.user.timeZoneCode"
    ],
    "targetPath": "$.timezone",
    "defaultValue": "Europe/Berlin",
    "valueMappings": [
        {
            "key": [
                "WDFt"
            ],
            "mappedValue": "Europe/Berlin"
        },
        {
            "key": [
                "ISRAEL"
            ],
            "mappedValue": "Asia/Jerusalem"
        },
        {
            "key": [
                "RUS03"
            ],
            "mappedValue": "Europe/Moscow"
        },
        {
            "key": [
                "AUSNSW"
            ],
            "mappedValue": "Australia/Sydney"
        },
        {
            "key": [
                "UTC+4"
            ],
            "mappedValue": "Asia/Dubai"
        },
        {
            "key": [
                "BRAZIL"
            ],
            "mappedValue": "America/Sao_Paulo"
        },
        {
            "key": [
                "BRZLEA"
            ],
            "mappedValue": "America/Sao_Paulo"
        },
        {
            "key": [
                "MSTNO"
            ],
            "mappedValue": "America/Phoenix"
        },
    ],
}

```

```

        "key": [
            "EST"
        ],
        "mappedValue": "America/New_York"
    },
    {
        "key": [
            "UTC"
        ],
        "mappedValue": "Etc/UTC"
    },
    {
        "key": [
            "UTC+3"
        ],
        "mappedValue": "Asia/Riyadh"
    },
    {
        "key": [
            "EST_"
        ],
        "mappedValue": "America/Toronto"
    },
    {
        "key": [
            "UTC+8"
        ],
        "mappedValue": "Asia/Shanghai"
    },
    {
        "key": [
            "JAPAN"
        ],
        "mappedValue": "Asia/Tokyo"
    }
]
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.businessPartnerRoleCode"
    ],
    "targetPath": "$.userType",
    "defaultValue": "Employee",
    "valueMappings": [
        {
            "key": [
                "BUP003"
            ],
            "mappedValue": "Employee"
        },
        {
            "key": [
                "BBP005"
            ],
            "mappedValue": "Contingent Worker"
        }
    ]
}
]
},
"group": {
    "mappings": [
        {
            "sourcePath": "$.ID",
            "targetPath": "$.displayName",
            "targetVariable": "entityIdSourceSystem",
            "functions": [

```

```

        {
          "condition": "'%s4hana.cloud.roles.prefix%' != 'null'",
          "function": "concatString",
          "prefix": "%s4hana.cloud.roles.prefix%"
        }
      ]
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.to_BusinessUserAssignment.results",
      "optional": true,
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.members"
    },
    {
      "type": "remove",
      "targetPath": "$.members[*].__metadata"
    },
    {
      "type": "rename",
      "constant": "value",
      "targetPath": "$.members[*].PersonID"
    },
    {
      "constant": "User",
      "targetPath": "$.members[*].type"
    }
  ]
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource. To learn how, see [Guided Answers: Business Role Description](#).

- Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA Cloud Documentation](#)

## 1.6.1.29 SAP S/4HANA for procurement planning

Follow this procedure to set up SAP S/4HANA for procurement planning as a source system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

You have technical credentials for SAP S/4HANA for procurement planning. See: [Onboarding](#)

### Context

SAP S/4HANA for procurement planning is a cloud-based solution designed to help you plan procurement activities with regard to the time schedule, as well as the investment planning of items based on a central bill of material.

You can use Identity Provisioning to configure SAP S/4HANA for procurement planning as a source system where you can read **users** from and provision them to a target system.

#### i Note

SAP S/4HANA for procurement planning does not support groups.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP S/4HANA Procurement Planning* as a source system. See: [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

#### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL to the SCIM API of your SAP S/4HANA for procurement planning system without path information.</p> <p>For example: <code>https://procplanning-api.cfapps.eu10.hana.ondemand.com</code></p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client Id, created for your SAP S/4HANA for procurement planning system.
Password	Enter the OAuth Client Secret, created for your SAP S/4HANA for procurement planning system.
OAuth2TokenServiceURL	<p>Enter the OAuth 2.0 Token Service URL.</p> <p>For example: <b><code>https://procplansecurity.authentication.eu10.hana.ondemand.com/oauth/token</code></b></p>
(Optional) <code>s4hana.pp.user.filter</code>	<p>When specified, only those SAP S/4HANA for procurement planning users matching the filter expression will be read.</p> <p>Example: <b><code>name.familyName eq "Smith" and addresses.country eq "US"</code></b></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP S/4HANA for procurement planning](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your system. For more information, see:

[Manage Transformations \[page 1494\]](#)

**Mapping logic** – the behavior of the default transformation logic is to read all user attributes from the source SAP S/4HANA for procurement planning system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

**Default transformation:**

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "type": "remove",
        "targetPath": "$.id"
      },
      {
        "type": "remove",
        "targetPath": "$.meta"
      }
    ]
  }
}
```

5. Add a target system to provision users to it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).



2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA for procurement planning – Product Page](#)

### 1.6.1.30 SAP S/4HANA On-Premise

Follow this procedure to set up SAP S/4HANA on-premise (also valid for SAP S/4HANA Cloud, private edition) as a source system.

## Prerequisites

### Note


If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)
- You have technical credentials (user and password) for SAP S/4HANA on-premise.
- The SAP S/4HANA on-premise system is version **1809** or higher.
- You have configured your SOA Manager to directly call the following Web services:
  - **ManageBusinessUserIn**
  - **QueryBusinessUserIn**

For more information, see: [Setting Up SOA Manager](#).

## Context

You can use Identity Provisioning to configure SAP S/4HANA on-premise as a source system where you can read **business users** (employee, collaboration user, contingent worker, and resource) from and provision them to a target system.

In SAP S/4HANA, a business user is defined as a natural person who is represented by a business partner and a link to a user in the system. Actually, the business user is an AS ABAP (SU01) user who also has a one-to-one relation to a corresponding business partner. For more information on the identity model for business users, see SAP Note [2570961](#) .

SAP S/4HANA on-premise supports reading of users with *User UUID* attribute which is generated by Identity Authentication at user creation. The attribute mapping is handled by the default transformation of SAP AS ABAP connector. Therefore, in order to read a user with *User UUID* from S/4HANA on-premise, the user should first be created (provisioned) in AS ABAP and then linked to its corresponding business user in S/4HANA on-premise. For more information, see: [SAP Application Server ABAP \[page 484\]](#).

Reading of **roles** (considered as *groups*) is also handled by the default transformation SAP AS ABAP connector.

## Procedure

1. Open the Cloud Connector to add an access control system mapping for **SAP S/4HANA On-Premise**. This is needed to allow the Identity Provisioning service to access SAP S/4HANA On-Premise as a back-end system on the intranet. To learn how, see: [Configure Access Control \(HTTP\)](#)
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add *SAP S/4HANA On-Premise* as a source system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to your SAP S/4HANA On-Premise system.
ProxyType	Enter: <i>OnPremise</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the technical user for SAP S/4HANA On-Premise.
Password	(Credential) Enter the password for the SAP S/4HANA On-Premise technical user.

Property Name	Description & Value
<code>s4hana.onprem.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source SAP S/4HANA On-Premise system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
<code>ips.date.variable.format</code>	<code>yyyy-MM-dd</code>
(Optional) <code>s4hana.onprem.sap-client</code>	<p>Use this property if you want to specify a particular AS ABAP client to use as the <b>sap-client</b> URL parameter.</p> <p>If this property is not specified, the URL will open your default AS ABAP client. To learn more, see: <a href="#">Specifying the Client</a></p> <p>For more information about <b>sap-client</b>, see: <a href="#">SAP URL Parameters</a></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=OnPremise
URL=http://aaa777.myhost:1234
User=MYS4HANAUSER
Password=*****
ips.date.variable.format=yyyy-MM-dd
s4hana.onprem.skip.read.archived=true
s4hana.onprem.sap-client=101
```

##### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP S/4HANA On-Premise](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in SAP S/4HANA On-Premise. For more information, see:

[Manage Transformations \[page 1494\]](#)

[APIs for Business User Management](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.validityPeriod.startDate <= '${currentDate}') && ($.validityPeriod.endDate > '${currentDate}')",
    "mappings": [
      {
        "sourcePath": "$.personID",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.personalInformation.firstName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.lastName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.personFullName",
        "targetPath": "$.name.formatted",
        "optional": true
      },
      {
        "sourcePath": "$.personalInformation.nickName",
        "targetPath": "$.nickName",
        "optional": true
      },
      {
        "sourcePath": "$.userAssignment.userID",
        "targetPath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "constant": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.workplaceInformation.emailAddress",
        "targetPath": "$.emails[0].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.PersonExternalID",
        "optional": true,
        "correlationAttribute": true
      }
    ]
  }
}
```

```

{
  "type": "valueMapping",
  "sourcePaths": [
    "$.businessPartnerRoleCode"
  ],
  "targetPath": "$.userType",
  "defaultValue": "Employee",
  "valueMappings": [
    {
      "key": [
        "BUP003"
      ],
      "mappedValue": "Employee"
    },
    {
      "key": [
        "BBP005"
      ],
      "mappedValue": "Contingent Worker"
    },
    {
      "key": [
        "BUP012"
      ],
      "mappedValue": "Collaboration User"
    },
    {
      "key": [
        "WFM001"
      ],
      "mappedValue": "Resource"
    }
  ]
}

```

6. Now, add a target system to provision users into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA On-Premise](#)  
[APIs for Business User Management](#)  
[Maintain Collaboration Users](#)

## 1.6.1.31 SAP Sales Cloud and SAP Service Cloud


Follow this procedure to set up SAP Sales Cloud and SAP Service Cloud, formerly known as [SAP Cloud for Customer](#) (in short, C4C), as a source system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

To integrate SAP Sales Cloud and SAP Service Cloud with Identity Provisioning, you need to use SAP Cloud Integration (SAP CI). This service provides a package with integration flows (iFlows) for enabling the creation of users and assignment of users to groups via SCIM API in SAP Sales Cloud and SAP Service Cloud.

- To configure SAP Cloud Integration and SAP Sales Cloud and SAP Service Cloud, see: [Identity Provisioning in SAP Cloud for Customer using System for Cross-Domain Identity Management \(SCIM\)](#)
- To set up and use the [SAP Cloud for Customer Integration with Identity Provisioning via System for Cross-domain Identity Management](#) package, see: [SAP Business Accelerator Hub: SAP Cloud for Customer Integration with Identity Provisioning via System for Cross-domain Identity Management](#) 

### Context

SAP Sales Cloud and SAP Service Cloud is a cloud-based solution that helps customers manage day-to-day sales and service interactions by sending and receiving signals between front- and back-office solutions and providing a single view of the customer.

You can use Identity Provisioning to configure SAP Sales Cloud and SAP Service Cloud as a source system where you can read [business users](#), [employee users](#), and [groups](#) from and provision them to a target system.

This scenario is relevant for existing SAP Sales Cloud and SAP Service Cloud customers (brown-field approach). This means that business users, employees, and groups have already been created in SAP Sales Cloud and SAP Service Cloud and can be provisioned to the Identity Authentication target system. After the users and groups are created in Identity Authentication, reflecting the business users and group assignments in SAP Sales Cloud and SAP Service Cloud, Identity Authentication can become the leading system in user provisioning.

### Procedure

1. Access the Identity Provisioning UI.

- [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Sales Cloud and SAP Service Cloud* as a source system. For more information, see [Add a System \[page 1477\]](#).
  3. Choose the *Properties* tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter SAP Cloud Integration runtime URL. See: <a href="#">How to Get SAP Cloud Integration Runtime URL</a>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter your authentication method: <i>BasicAuthentication</i>
User	Enter SAP Cloud Integration user ID to connect to SAP Cloud Integration. See: <ul style="list-style-type: none"> <li>• <a href="#">Setting Up Inbound HTTP Connections (with Basic Authentication), Neo Environment</a></li> <li>• <a href="#">Basic Authentication of IdP User for Integration Flow Processing (Cloud Foundry environment)</a></li> </ul>
Password	(Credential) Enter SAP Cloud Integration password to connect to SAP Cloud Integration. See: <ul style="list-style-type: none"> <li>• <a href="#">Setting Up Inbound HTTP Connections (with Basic Authentication), Neo Environment</a></li> <li>• <a href="#">Basic Authentication of IdP User for Integration Flow Processing (Cloud Foundry environment)</a></li> </ul>
c4c.api.version	The version of the SAP Sales Cloud and SAP Service Cloud API you use. By default, the Identity Provisioning service uses version <b>3</b> - the SCIM 2.0 based API.

Property Name	Description & Value
(Optional) <code>c4c.user.filter</code>	<p>When specified, only those C4C users matching the filter expression will be read.</p> <p>For example:</p> <ul style="list-style-type: none"> <li>• <code>userName eq "Smith"</code></li> <li>• <code>email eq "test@abc.com"</code></li> <li>• <code>employeeNumber eq "56789"</code></li> <li>• <code>addresses.country eq "USA"</code></li> </ul>
(Optional) <code>c4c.group.filter</code>	<p>When specified, only those C4C groups matching the filter expression will be read.</p> <p>Example: <b><code>displayName eq "ProjectTeam1"</code></b></p>
(Optional) <code>c4c.group.prefix</code>	<p>This property distinguishes SAP Sales Cloud and SAP Service Cloud groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b><code>C4C_</code></b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Sales Cloud and SAP Service Cloud source system and will be provisioned to the target system with the following name pattern: <b><code>C4C_&lt;GroupDisplayName&gt;</code></b>. This way SAP Sales Cloud and SAP Service Cloud groups in the target system will be distinguished from groups provisioned from other applications.</p> <p>If the property is not set, the SAP Sales Cloud and SAP Service Cloud groups will be read and provisioned to the target system with their actual display names.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

The Identity Provisioning offers a default transformation for the *SAP Sales Cloud and SAP Service Clouds* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in C4C. For more information, see [Manage Transformations \[page 1494\]](#).

#### Default transformation:



## Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.userType",
        "optional": true,
        "targetPath": "$.userType"
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.middleName",
        "optional": true,
        "targetPath": "$.name.middleName"
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath":
          "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
          ['employeeNumber']",
        "targetPath":
          "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
          ['employeeNumber']"
      },
      {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "functions": [
          {
            "condition": "'%c4c.group.prefix%' != 'null'",
```

```

        "function": "concatString",
        "applyOnElements": true,
        "applyOnAttribute": "display",
        "prefix": "%c4c.group.prefix%"
      }
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "'%c4c.group.prefix%' != 'null'",
            "function": "concatString",
            "prefix": "%c4c.group.prefix%"
          }
        ]
      }
    ],
    "sourcePath": "$.members",
    "preserveArrayWithSingleElement": true,
    "targetPath": "$.members",
    "optional": true
  },
  {
    "sourcePath": "$.schemas",
    "preserveArrayWithSingleElement": true,
    "targetPath": "$.schemas"
  }
]
}

```

5. Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.32 SAP SuccessFactors

Follow this procedure to set up SAP SuccessFactors as a source system.

### Prerequisites

You have created a technical user with permissions to **call** the SAP SuccessFactors HCM Suite OData API and to **export** employee data from the SAP SuccessFactors system. You need the following mandatory permissions:

- [Manage User](#) > [Employee Export](#)
- [Manage User](#) > [User Account OData API Entity](#)
- [Manage Integration Tools](#) > [Allow Admin to Access OData API](#)
- [Manage Role-Based Permission Access](#) > [Role-Based Permission Admin](#)
- [Admin Center](#) > [Manage Permission Roles](#) > [Access to X.509 Certificates](#) permission (needed for configuring X.509 certificate-based authentication)

For more information, see [Permissions](#) and [Setting Up an API User for Sync Jobs](#).

### Context

Companies that manage their employees using SAP SuccessFactors HCM Suite can use the Identity Provisioning service to automatically create accounts for these employees and manage their permissions for the cloud applications. When the hiring process of a new employee is completed in the SAP SuccessFactors HCM solution, a user record with the employee identity data is created in the SAP SuccessFactors system. The Identity Provisioning service can use this data for the identity and authorization provisioning processes.

SAP SuccessFactors provides two APIs for its integration with Identity Provisioning: SAP SuccessFactors HCM Suite OData API and SAP SuccessFactors Workforce SCIM API. The value of `sf.api.version` property controls which API you use.

- When the value is set to **1**, or the property is not defined - SAP SuccessFactors HCM Suite OData API (in short, OData API) is used. This is the default value. SAP SuccessFactors source systems created before the introduction of `sf.api.version` property, use OData API.
- When the value is set to **2** - SAP SuccessFactors Workforce SCIM API (in short, SCIM API) is used. For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#).

Identity Provisioning supports reading of SAP SuccessFactors users, as well as static and dynamic groups of type Permission.

When users from SAP SuccessFactors source system are provisioned to Identity Authentication target system, all newly created users get the `userUUID` attribute generated by Identity Authentication. The Identity Provisioning in its turn sends the `userUUID` back to SAP SuccessFactors to update the users.




To create SAP SuccessFactors as a source system, proceed as follows:

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP SuccessFactors* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP SuccessFactors and configure the authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP SuccessFactors source system, select the *Certificate* tab and choose  *Generate*  *Download* , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip this step if you use basic authentication. The next steps are performed in SAP SuccessFactors Admin Center and are relevant for certificate-based authentication only.

- b. Login to SAP SuccessFactors and go to *Admin Center*. Follow the procedure described in [Upgrade to X.509 Certificate-Based Authentication for Incoming Calls](#).

Make sure you select *Identity Provisioning Service* in the *Integration Name* field.

4. Choose the *Properties* tab to configure the connection settings for your system.

### i Note


If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>

Property Name	Description & Value
URL	<p>Specify the URL to your SAP SuccessFactors API.</p> <p>For example:</p> <ul style="list-style-type: none"> <li>For version 1: <a href="https://apitest.successfactors.com/odata/v2">https://apitest.successfactors.com/odata/v2</a></li> <li>For version 2: <a href="https://apitest.successfactors.com">https://apitest.successfactors.com</a></li> </ul> <p>To see the list of all SAP SuccessFactors data centers, see: <a href="#">HXM Suite OData APIs: API Endpoint URLs</a> and <a href="#">System for Cross-domain Identity Management for Workforce in SuccessFactors</a> </p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"> <li><a href="#">BasicAuthentication</a></li> <li><a href="#">ClientCertificateAuthentication</a></li> </ul>
(Optional) <code>sf.api.version</code>	<p>Handles the version of the API which is consumed by the SAP SuccessFactors system.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">1</a> - Indicates that SAP SuccessFactors HCM Suite OData API (in short, OData API) is used.</li> <li><a href="#">2</a> - Indicates that SAP SuccessFactors Workforce SCIM API (in short, SCIM API) is used.</li> </ul> <p>Default value: <a href="#">1</a></p>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">userID</a> of your SAP SuccessFactors technical user in the following format: <code>&lt;user_ID&gt;@&lt;company_ID&gt;</code></p>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the password for your SAP SuccessFactors technical user.</p>

Property Name	Description & Value
<code>sf.company.id</code>	<p>Valid if <a href="#">ClientCertificateAuthentication</a> is configured as authentication method.</p> <p>Enter the Company ID of your SAP SuccessFactors system.</p> <p>The Company ID is a short string of characters that identifies each SAP SuccessFactors system. It is like a user-name for your organization. All users of the same system share the same Company ID.</p>
<code>sf.user.attributes</code>	<p>Default property. It's a string representing a comma-separated list of user attributes that have to be loaded (read) from SAP SuccessFactors. You can leave the default property value (all listed attributes), leave only some of them, or add some more.</p> <div> <p><b>Note</b></p> <p>If you want to add more attributes, make sure you have added:</p> <ul style="list-style-type: none"> <li>the relevant extra attributes to the value of this property, separated by commas</li> <li>extra mappings for these attributes in the <a href="#">user</a> transformation</li> <li>extra mappings for these attributes in the write transformation of the relevant target system</li> </ul> </div> <div> <p><b>→ Remember</b></p> <ul style="list-style-type: none"> <li>Always make sure that attribute <code>lastModifiedDateTime</code> is in the list of values. If you delete it (don't specify it), the provisioning from SAP SuccessFactors will fail.</li> <li>If a user in SAP SuccessFactors is missing the <code>lastModifiedDateTime</code> attribute, it will break the provisioning. As a solution, you can exclude such users from the provisioning – either by using the <code>sf.user.filter</code> property, or by setting a condition in the transformation logic.</li> </ul> </div> <p><b>Connector version:</b> SAP SuccessFactors version 1</p>

Property Name	Description & Value
(Optional) <code>sf.user.attributes.expand</code>	<p>This property reads additional user data related to <a href="#">complex (navigation) attributes</a>, which are specified in <code>sf.user.attributes</code>.</p> <p>Default value: <a href="#">personKeyNav</a>, <a href="#">personKeyNav/</a> <a href="#">userAccountNav</a></p> <div> <p><b>Note</b></p> <p>If you want to add more complex attributes, enter the correct "paths" to them, specifying all parent attribute levels.</p> <p>For example, if you want to read information about user's skills and competency, add <a href="#">positionSkillMappings/skill</a>. In this case, make sure you have added:</p> <ul style="list-style-type: none"> <li>• <b>positionSkillMappings</b> to the value of property <code>sf.user.attributes</code></li> <li>• a relevant mapping for this attribute in the <a href="#">user</a> transformation</li> <li>• a relevant mapping for this attribute in the write transformation of the relevant target system</li> </ul> </div> <p><b>Connector version:</b> SAP SuccessFactors version 1</p>
(Optional) <code>sf.user.filter</code>	<p>The possible values of this property depend on the API version which your SAP SuccessFactors system consumes.</p> <p>Use this property to filter users from SAP SuccessFactors. The filter obtains values as described in the OData 2.0 syntax, except any statements with attribute <code>lastModifiedDateTime</code>.</p> <p>Find below example syntax for filtering users depending of the API version:</p> <ul style="list-style-type: none"> <li>• OData API - <code>username eq 'cbraun'</code></li> <li>• SCIM API - <code>userName eq "cbraun"</code></li> </ul> <p>To learn more, see:</p> <ul style="list-style-type: none"> <li>• <a href="#">OData version 2</a> → <b>4.5. Filter System Query Option (\$filter)</b>.</li> <li>• <a href="#">SAP SuccessFactors HXM Suite OData API: Reference Guide (V2)</a></li> <li>• <a href="#">SAP SuccessFactors Workforce SCIM API and System for Cross-domain Identity Management for Workforce in SuccessFactors</a></li> </ul>

Property Name	Description & Value
(Optional) <code>sf.user.read.deactivatedafter</code>	<p>This property filters SAP SuccessFactors inactive users from a particular date on. It is an optional property which does not appear by default at system creation. It accepts a value in the <b>yyyy-MM-dd</b> format. For example: <b>2023-07-17</b></p> <p>The <code>sf.user.read.deactivatedafter</code> property works together with the <code>sf.user.filter</code> property which is added at system creation with the default value: <code>active eq true</code>. Using it can further narrow down the filtering results.</p> <p>To filter active users along with inactive ones from a particular date on, the following configuration must be in place:</p> <ul style="list-style-type: none"> <li>Set the <code>sf.user.read.deactivatedafter</code> value to a date in the expected format. For example: <code>2023-07-17</code></li> <li><code>sf.user.filter = active eq true</code></li> </ul> <p>As a result, Identity Provisioning reads SAP SuccessFactors active users and the users set to inactive from that date on using the 2023-07-17T00:00:00Z date-time format.</p> <p>Depending on the value you define for <code>sf.user.filter</code>, expect the following results:</p> <ul style="list-style-type: none"> <li><code>sf.user.filter = active eq false</code> All inactive users will be returned.</li> <li><code>sf.user.filter = active eq false and userName sw "Test_"</code> All inactive users with username starting with Test_ will be returned.</li> </ul> <div> <p><b>Note</b></p> <p>When you filter by <code>sf.user.filter = active eq false</code> along with the property <code>sf.user.read.deactivatedafter</code>, the users that match the two criteria will be read twice.</p> </div> <ul style="list-style-type: none"> <li><code>sf.user.filter = active eq true and userName sw "Test_"</code> Inactive users from the provided date on and all active users with username starting with Test_ will be returned.</li> </ul>



Property Name	Description & Value
	<b>Connector version:</b> SAP SuccessFactors version 2
(Optional) <code>sf.group.filter</code>	<p>The possible values of this property depend on the API version which your SAP SuccessFactors system consumes.</p> <p>Use this property to filter dynamic groups from SAP SuccessFactors. The filter obtains values as described in the OData 2.0 syntax, except any statements with attribute <code>lastModifiedDateTime</code>. To learn more, see:</p> <ul style="list-style-type: none"> <li>• <a href="#">OData version 2</a> → <b>4.5. Filter System Query Option (\$filter)</b></li> <li>• <a href="#">SAP SuccessFactors HXM Suite OData API: Reference Guide (V2)</a> → <b>DynamicGroup</b></li> <li>• <a href="#">SAP SuccessFactors Workforce SCIM API and System for Cross-domain Identity Management for Workforce in SuccessFactors</a></li> </ul>
(Optional) <code>sf.group.prefix</code>	<p>This property distinguishes SAP SuccessFactors groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SF_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP SuccessFactors source system and will be provisioned to the target system with the following name pattern: <b>SF_&lt;GroupDisplayName&gt;</b>. This way SAP SuccessFactors groups in the target system will be distinguished from groups provisioned from other applications.</p> <p>If the property is not set, the SAP SuccessFactors groups will be read and provisioned to the target system with their actual display names.</p> <p><b>Connector version:</b> SAP SuccessFactors version 2</p>
(Optional) <code>sf.page.size</code>	<p>Defines the paging size.</p> <ul style="list-style-type: none"> <li>• Default value: <b>100</b></li> <li>• Maximum value: <b>1000</b></li> </ul> <p><b>Connector version:</b> SAP SuccessFactors version 1</p>

Property Name	Description & Value
(Optional)sf.group.members.paging.enabled	<p>This property enables paging of group members.</p> <p>The maximum number of group members returned per request is 100. To read more than 100 group members, paging must be enabled.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i> - Paging is enabled.</li> <li><i>false</i> - Paging is disabled.</li> </ul> <p>Default value: <i>false</i></p> <p><b>Connector version:</b> SAP SuccessFactors version 2</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination (configuration):

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://apitest.successfactors.com/odata/v2
User=sfsf_admin@mycompany.com
Password=*****
sf.user.attributes=userId,username,status,email,lastName,firstName,lastModifiedDateTime,personKeyNav,positionSkillMappings
sf.user.attributes.expand=personKeyNav,personKeyNav/userAccountNav,positionSkillMappings/skill
sf.user.filter=department ne 'Manufacturing'
sf.group.filter=groupType eq 'permission'
sf.page.size=70
```

##### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP SuccessFactors* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

When the SAP SuccessFactors system is configured as a source, the Identity Provisioning service will read all the attributes of the user records supported by the SAP SuccessFactors API.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP SuccessFactors. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP SuccessFactors HCM Suite OData API](#)

[SAP SuccessFactors HXM Suite OData API: Reference Guide \(V2\)](#)

[SAP SuccessFactors Workforce SCIM API](#)

#### Default transformation for SAP SuccessFactors HCM Suite OData AP version 1:

##### Code Syntax

```
// The value of entityIdSourceSystem is used for storing the unique ID of
// each user.
// You can change the default source attribute perPersonUuid, but make
// sure the new source attribute is also unique.
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.personKeyNav.perPersonUuid",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "constant": false,
        "targetPath": "$.active"
      },
      {
        "condition": "$.personKeyNav.userAccountNav.accountStatus
== 'ACTIVE'",
        "constant": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.personKeyNav.userAccountNav.username",
        "targetPath": "$.userName"
      },
      {
        "constant": "employee",
        "targetPath": "$.userType"
      },
      {
        "sourcePath": "$.firstName",
        "optional": true,
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.lastName",
        "targetPath": "$.name.familyName"
      }
    ],
    // The email attribute is used as a first value for the emails array of
    // the intermediate JSON data.
    {
      "sourcePath": "$.email",
      "targetPath": "$.emails[0].value"
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
      "targetPath": "$.schemas[0]"
    },
    {
      "constant": "urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User",
```

```

        "targetPath": "$.schemas[1]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": "$.schemas[2]"
      },
      {
        "sourcePath": "$.personKeyNav.perPersonUuid",
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['personGUID']"
      },
      {
        "sourcePath": "$.userId",
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:sfsf:2.0:User']['userSysID']"
      },
      {
        "sourcePath": "$.personKeyNav.personIdExternal",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']"
      }
    ]
  },
  // By default, the group mapping is inactive (ignored) but groups are
  // supported.
  // To start provisioning groups, either delete the statement "ignore":
  // true, or set its value to false.
  "group": {
    "ignore": true,
    "mappings": [
      {
        "sourcePath": "$.groupID",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.groupName",
        "targetPath": "$.displayName"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath": "$.schemas[0]"
      },
      {
        "sourcePath": "$.users[*].personGUID",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]"
      }
    ]
  }
}

```

## Default transformation for SCIM API version 2:

### Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.schemas",

```

```

    "preserveArrayWithSingleElement": true,
    "targetPath": "$.schemas"
  },
  {
    "sourcePath": "$.id",
    "targetVariable": "entityIdSourceSystem"
  },
  {
    "sourcePath": "$.userName",
    "targetPath": "$.userName",
    "correlationAttribute": true
  },
  {
    "sourcePath": "$.userType",
    "targetPath": "$.userType"
  },
  {
    "sourcePath": "$.name.familyName",
    "optional": true,
    "targetPath": "$.name.familyName"
  },
  {
    "sourcePath": "$.name.givenName",
    "optional": true,
    "targetPath": "$.name.givenName"
  },
  {
    "sourcePath": "$.name.middleName",
    "optional": true,
    "targetPath": "$.name.middleName"
  },
  {
    "sourcePath": "$.name.honorificPrefix",
    "optional": true,
    "targetPath": "$.name.honorificPrefix"
  },
  {
    "sourcePath": "$.name.honorificSuffix",
    "optional": true,
    "targetPath": "$.name.honorificSuffix"
  },
  {
    "sourcePath": "$.name.formatted",
    "optional": true,
    "targetPath": "$.name.formatted"
  },
  {
    "sourcePath": "$.nickName",
    "optional": true,
    "targetPath": "$.nickName"
  },
  {
    "sourcePath": "$.preferredLanguage",
    "optional": true,
    "targetPath": "$.preferredLanguage"
  },
  {
    "sourcePath": "$.displayName",
    "optional": true,
    "targetPath": "$.displayName"
  },
  {
    "sourcePath": "$.title",
    "optional": true,
    "targetPath": "$.title"
  },
  {
    "sourcePath": "$.externalId",

```

```

        "optional": true,
        "targetPath": "$.externalId"
    },
    {
        "sourcePath": "$.locale",
        "optional": true,
        "targetPath": "$.locale"
    },
    {
        "sourcePath": "$.timezone",
        "optional": true,
        "targetPath": "$.timezone"
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['perPersonUuid']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['perPersonUuid']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['loginMethod']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['loginMethod']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['personIdExternal']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['personIdExternal']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['customFields']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['customFields']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']"
    },
    {

```

```

        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['division']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['$ref']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['$ref']"
    },
    {
        "sourcePath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.phoneNumbers"
    },
    {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails"
    },
    {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "optional": true,
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active"
    },
    {
        "sourcePath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups",
        "functions": [
            {
                "condition": "'%sf.group.prefix%' != 'null'",
                "function": "concatString",
                "applyOnElements": true,
                "applyOnAttribute": "display",
                "prefix": "%sf.group.prefix%"
            }
        ]
    }

```

```

    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "'%sf.group.prefix%' != 'null'",
            "function": "concatString",
            "prefix": "%sf.group.prefix%"
          }
        ]
      }
    ]
  },
  {
    "sourcePath": "$.members",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members"
  }
]
}

```

### Note

When SAP SuccessFactors users are provisioned to Identity Authentication, the service generates a user UUID for every newly created user. Within the same provisioning job, Identity Provisioning returns the user UUID back to SAP SuccessFactors and updates the *sapGlobalUserId* attribute.

Identity Provisioning updates *sapGlobalUserId* when its value is **null** or when its value already exists. The latter is valid in cases where, for example, a user is deleted in the Identity Authentication target system and after running a Resync Job, the deleted user is created again. A new user UUID is generated which updates the old value in SAP SuccessFactors.

Returning the user UUID back to SAP SuccessFactors and updating the *sapGlobalUserId* attribute is supported for standard and real-time provisioning scenarios. It is also supported in bulk scenarios, when bulk operations are enabled on the Identity Authentication target system.

In case you encounter error **statusCode 432** during userUUID synchronization, see: [Guided Answers: Error statusCode: 432](#) 📖


- Now, add a target system to provision users into it. Choose from: [Target Systems \[page 702\]](#)

### ⚠ Caution

The **email** attribute is not unique for *SAP SuccessFactors* but it's unique for systems like *Identity Authentication* or *SAP Analytics Cloud*. That's why, if you provision users to any of these systems, make



sure that there are no [SAP SuccessFactors](#) users with duplicate e-mails. If there are such, then in the target system all affected users will be created as a single user, with merged user data.

To learn more, see [Guided Answers: Multiple Users from a Source System Are Created as One in the Target](#) .

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[URI Conventions \(OData Version 2.0\)](#) 

[SAP SuccessFactors HCM Suite OData API](#)

[SAP SuccessFactors Workforce SCIM API](#)

[\(Guided Answers\) Multiple Users from SAP SuccessFactors Are Created as One in Identity Authentication](#) 

## 1.6.1.33 SAP SuccessFactors Learning

Follow this procedure to set up SAP SuccessFactors Learning as a source system.

## Prerequisites

You have created a technical user with administrator permissions that will be used to call the API of SAP SuccessFactors Learning for reading user information.

## Context

SAP SuccessFactors Learning is a learning solution which helps organizations to improve employee skills and talent management, align learning outcomes with performance goals, boost compliance, and train external audiences.

You can use Identity Provisioning to configure SAP SuccessFactors Learning as a source system where you can read **users** from and provision them to a target system.

When users from SAP SuccessFactors Learning source system, are provisioned to Identity Authentication target system, all newly created users get the *userUUID* attribute generated by Identity Authentication. The Identity Provisioning in its turn sends the *userUUID* back to SAP SuccessFactors Learning to update the users.

### ! Restriction

Reading *groups* is not supported.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP SuccessFactors Learning* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the API of your SAP SuccessFactors Learning system. It follows the pattern: <code>https://&lt;root URL&gt;/learning/public-api/rest/admin/Integration.svc/ias</code>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the technical user ID for SAP SuccessFactors Learning.

Property Name	Value
Password	Enter the password for the SAP SuccessFactors Learning technical user. For more information, see <a href="#">Learning Technical User</a> .
(Optional) <code>lms.user.filter</code>	<p>When specified, only those users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <code>userName eq "testName"</code></li> <li>• <code>externalID eq "testID"</code></li> <li>• <code>active eq "true"</code></li> <li>• <code>sourceSystem eq "Learning"</code> - indicates that the user is created directly in SAP SuccessFactors Learning with no involvement of Identity Provisioning.</li> <li>• <code>sourceSystem eq "Identity Provisioning"</code> - indicates that the user is created in SAP SuccessFactors Learning by Identity Provisioning.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

The Identity Provisioning offers a default transformation for the *SAP SuccessFactors Learning* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

The behavior of the default transformation logic is to read all user attributes from the source SAP SuccessFactors Learning system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

You can change the default transformation mapping rules depending on your setup of entities in your SAP SuccessFactors Learning. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: SAP SuccessFactors Learning](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      }
    ]
  }
}
```

```

        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
    },
    {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
    },
    {
        "sourcePath": "$.externalId",
        "optional": true,
        "targetPath": "$.externalId"
    },
    {
        "sourcePath": "$.locale",
        "optional": true,
        "targetPath": "$.locale"
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['siteID']",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['siteID']"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['sourceSystem']",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['sourceSystem']"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['applicationID']",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['applicationID']"
    },
    {
        "sourcePath": "$['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['110']",
        "optional": true,
        "targetPath": "$.custom-column-path-1"
    },
    {
        "sourcePath": "$['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['120']",
        "optional": true,
        "targetPath": "$.custom-column-path-2"
    },
    {
        "sourcePath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,

```

```

        "optional": true,
        "targetPath": "$.phoneNumbers"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails"
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      }
    ]
  }
}

```

### → Tip

Your main scenario with this source connector might be provisioning SAP SuccessFactors Learning users to SAP Cloud Identity Service – Identity Authentication. In this case, you can add the following **condition** at the beginning of the "user" resource since [emails](#) and [name.familyName](#) are mandatory attributes for Identity Authentication users.

```

{
  "user": {
    "condition": "($.emails[0].value EMPTY false) && ($.name.familyName EMPTY false)",
    "mappings": [
      ...
    ]
  }
}

```

### i Note

When SAP SuccessFactors Learning users are provisioned to Identity Authentication, the service generates a user UUID for every newly created user. Within the same provisioning job, Identity Provisioning returns the user UUID back to SAP SuccessFactors Learning and updates the [userUUID](#) attribute.

Identity Provisioning updates [userUUID](#) when its value is [null](#) or when its value already exists. The latter is valid in cases where, for example, a user is deleted in the Identity Authentication target system and after running a Resync Job, the deleted user is created again. A new user UUID is generated which updates the old value in SAP SuccessFactors Learning.

Returning the user UUID back to SAP SuccessFactors Learning and updating the [userUUID](#) attribute is supported for standard and real-time provisioning scenarios. It is also supported in bulk scenarios, when bulk operations are enabled on the Identity Authentication target system.

5. Now, add a target system to provision users to it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP SuccessFactors Learning](#) 

### 1.6.1.34 Sales Cloud – Analytics & AI

Follow this procedure to set up Sales Cloud – Analytics & AI as a source system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

You have technical user credentials for an Sales Cloud – Analytics & AI (in short, [SCAAI](#)) system with read and write access permissions.

## Context

After fulfilling the prerequisites, follow the procedure to add a source system for Sales Cloud – Analytics & AI to read users and user assignments to groups. This source system consumes SCIM 2.0 API provided by SCAAI.

## Procedure

1. Access the Identity Provisioning UI.

- [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *Sales Cloud – Analytics & AI* as a source system. For more information, see [Add a System \[page 1477\]](#).
  3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the SCIM API portal of your SCAA system.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user for your SCAA system.
Password	Enter the password for your SCAA user.
OAuth2TokenServiceURL	Enter the URL to the OAuth2 token service.  If not sure about the exact URL, ask your SCAA administrator.
(Optional)sales.cloud.analytics_ai.group.filter	Enter a group filter criteria, according to the API syntax of SCAA.  For example: <b>displayName eq "first_group"</b>
(Optional)sales.cloud.analytics_ai.user.filter	Enter a user filter criteria, according to the API syntax of SCAA.  For example: <b>externalId eq "John123"</b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

Type=*HTTP*

Authentication=*BasicAuthentication*

ProxyType=*Internet*

URL=*http://myscaai:8080/scim\_services*

User=*MySCAAIUser*

Password=*\*\*\*\*\**

OAuth2TokenServiceURL=*http://myscaai:8080/gateway\_services/api/auth/ips/token*

---

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Sales Cloud – Analytics & AI* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SCAAI system. For more information, see [Manage Transformations \[page 1494\]](#).

The behavior of the default transformation logic is to read all user attributes from the source SCAAI system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "type": "remove",
        "targetPath": "$.groups[*].display"
      }
    ]
  }
}
```



```

        {
            "type": "remove",
            "targetPath": "$.groups[*].ref"
        }
    ],
    "group": {
        "mappings": [
            {
                "sourcePath": "$.id",
                "targetVariable": "entityIdSourceSystem"
            },
            {
                "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
                "targetPath": "$.schemas[0]"
            },
            {
                "sourcePath": "$.displayName",
                "targetPath": "$.displayName"
            },
            {
                "optional": true,
                "preserveArrayWithSingleElement": true,
                "sourcePath": "$.members",
                "targetPath": "$.members"
            },
            {
                "type": "remove",
                "targetPath": "$.members[*].$ref"
            },
            {
                "type": "remove",
                "targetPath": "$.members[*].display"
            }
        ]
    }
}

```

5. Now, add a target system to provision users and their group assignments to it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.35 Cloud Foundry UAA Server

Follow this procedure to set up the Cloud Foundry UAA server as a source system.

### Prerequisites

#### ! Restriction

This system is available for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- You have technical user credentials for a Cloud Foundry system with read access permissions. In case OAuth is used for authentication, client ID and secret are required when creating a destination for access token retrieval. You need Cloud Foundry UAA version **4.2** or higher.
- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need to do this only if the Cloud Foundry UAA server is exposed in a private corporate network. For more information, see [Cloud Connector](#).

### Context

User Account and Authentication Service (UAA) is an OAuth2 server that you can use for centralized identity management. It owns the user accounts and authentication sources, and supports standard protocols (such as [SAML](#), [LDAP](#), and [OpenID Connect](#)) to provide SSO and delegated authorization to Web applications. For more information, see [Cloud Foundry: Overview](#) .

Cloud Foundry UAA is responsible for the SAP ID service to create and manage platform users (platform administrators and platform developers) in Cloud Foundry.

#### → Tip

This connector is meant for reading users and groups from **general** Cloud Foundry systems (they could be non-SAP ones). If you want to trigger provisioning of entities from SAP Business Technology Platform Cloud Foundry applications, you'd better use [SAP BTP XS Advanced UAA \(Cloud Foundry\) \[page 519\]](#) source system.

These source systems consume SCIM 1.1 API provided by Cloud Foundry UAA.

#### → Remember

You can read Cloud Foundry users and groups on **application** level only. You cannot read or manage them on a [subaccount](#) level.

Follow the steps below to create Cloud Foundry UAA as a source system to read users and groups, which can then be provisioned to a certain target system.

## Procedure

1. (Optional) If the Cloud Foundry UAA server is exposed in a private corporate network, add an access control system mapping in Cloud Connector. For more information, see [Configure Access Control \(HTTP\)](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add *Cloud Foundry UAA Server* as a source system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the Cloud Foundry UAA SCIM API.  If not sure about the exact URL, ask your Cloud Foundry UAA administrator.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
OAuth2TokenServiceURL	As you need to make OAuth authentication to the UAA system, enter the URL to the OAuth2 token service.  If not sure about the exact URL, ask your Cloud Foundry UAA administrator.
User	Enter the OAuth client ID of the Cloud Foundry UAA technical user.
Password	Enter the OAuth client secret of the technical user.

Property Name	Description & Value
uaa.origin	<p>Enter the location of your Cloud Foundry identity provider. If not sure about the value, ask your Cloud Foundry UAA administrator.</p> <p>The value of this property is a string, which will be used as the <i>origin</i> attribute in the system transformation.</p>
uaa.origin.filter.enabled	<p>This flag property depends on <code>uaa.origin</code>. Possible values: <b>true</b> or <b>false</b></p> <ul style="list-style-type: none"> <li>• If set to <i>true</i>, the Identity Provisioning service will read only users whose identity provider is set as a value of <code>uaa.origin</code>.</li> <li>• If set to <i>false</i>, the Identity Provisioning service will read all users, regardless of their origin.</li> <li>• If set to <i>true</i> but the <code>uaa.origin</code> property is missing, the provisioning job will fail.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

Type=*HTTP*

Authentication=*BasicAuthentication*

ProxyType=*Internet*

URL=*https://api.authentication.hana.ondemand.com*

OAuth2TokenServiceURL=*https://MyCFaccount.authentication.hana.ondemand.com/oauth/token*

User=*MyCFuser*

Password=*\*\*\*\*\**

uaa.origin=*my\_UAA\_location*

uaa.origin.filter.enabled=*true*

## 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Cloud Foundry UAA Server* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

- **Mapping logic** – The behavior of the default transformation logic is to map all attributes from a Cloud Foundry UAA entity to the intermediate Identity Provisioning representation.
- **User offboarding** – If a user or group has been deleted from the Cloud Foundry UAA server, this change is recognized, and the user/group is deleted from the target system too.

You can change the default transformation mapping rules to reflect your current setup of entities in your Cloud Foundry UAA server. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Cloud Foundry UAA API: Users](#) ➡

[Cloud Foundry UAA API: Groups](#) ➡

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.verified",
        "targetPath": "$.verified",

```

```

        "optional": true
      },
      {
        "sourcePath": "$.meta",
        "targetPath": "$.meta",
        "optional": true
      },
      {
        "sourcePath": $.externalId,
        "targetPath": $.externalId,
        "optional": true
      },
      {
        "sourcePath": $.origin,
        "targetPath": $.origin,
        "optional": true
      },
      {
        "sourcePath": $.zoneId,
        "targetPath": $.zoneId,
        "optional": true
      },
      {
        "sourceVariable": "entityBaseLocation",
        "targetPath": $.meta.location,
        "targetVariable": "entityLocationSourceSystem",
        "functions": [
          {
            "type": "concatString",
            "suffix": "${entityIdSourceSystem}"
          }
        ]
      }
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": $.id,
        "targetPath": $.id,
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": $.displayName,
        "targetPath": $.displayName,
        "functions": [
          {
            "condition": "'%uua.group.prefix%' != 'null'",
            "function": "concatString",
            "prefix": "%uua.group.prefix%"
          }
        ]
      }
    ]
  },
  {
    "sourcePath": $.description,
    "targetPath": $.description,
    "optional": true
  },
  {
    "sourcePath": $.zoneId,
    "targetPath": $.zoneId,
    "optional": true
  },
  {
    "sourcePath": $.meta,
    "targetPath": $.meta,
    "optional": true
  },
}

```

```

    {
      "sourcePath": "$.members",
      "targetPath": "$.members",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "sourceVariable": "entityBaseLocation",
      "targetPath": "$.meta.location",
      "targetVariable": "entityLocationSourceSystem",
      "functions": [
        {
          "type": "concatString",
          "suffix": "${entityIdSourceSystem}"
        }
      ]
    }
  ]
}

```

6. Now, add a target system to which to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Cloud Foundry UAA: Users](#) ➡

[Cloud Foundry UAA: Groups](#) ➡

## 1.6.1.36 Google G Suite

Follow this procedure to set up Google G Suite as a source system.

## Prerequisites

1. Sign in to the Google API console (<https://console.developers.google.com> ➡) and create a project.

2. Enable the Admin SDK. To do this, go to [Dashboard](#) > [ENABLE API](#) > [Admin SDK](#) > [ENABLE](#).
3. Create a service account for your project. We recommend that you select [Enable G Suite Domain-wide Delegation](#) during the creation. If you skip this option, you can set it later. For more information, see [Creating a service account](#).
4. Then, in the Google admin console (<https://admin.google.com>), a user with **Super Admin** role can delegate domain-wide authority to your service account. This way, it will have access to the Google Admin SDK on behalf of your user. For more information, see [Delegating domain-wide authority](#).

### Note

When specifying the scopes, the administrator has to enter the following:

<https://www.googleapis.com/auth/admin.directory.user>, <https://www.googleapis.com/auth/admin.directory.group>

## Context

A Google service account with delegated domain-wide authority is required for authentication and authorization of the Identity Provisioning service to G Suite domain. The authentication is based on OAuth 2.0 protocol with JSON Web Token (JWT). The private key for the signature is distributed by Google via one-time downloadable JSON data, which is accessible by the domain administrator. The private key is encoded in PKCS8 format and is in the *private\_key* field of the JSON data. For more information, see [JSON Web Token \(JWT\)](#).

- When using it as a source system, you can read both users and groups from Google G Suite and provision them to any target system you have added in the Identity Provisioning user interface.
- When using it as a target system, you can write both users and groups, read from any source system you have added in the Identity Provisioning user interface. Google G Suite can automatically create accounts for your users in the Google Cloud Datastore.

The Identity Provisioning service supports user and group operations based on the following Google Directory API. See the table below.

User Operations	Group Operations
<a href="#">Create a user</a>	<a href="#">Create a group</a>
<a href="#">Retrieve a user</a>	<a href="#">Retrieve a group's properties</a>
<a href="#">Update a user</a>	<a href="#">Update a group's properties</a>
<a href="#">Delete a user</a>	<a href="#">Delete a group</a>

### Caution

You can only provision users whose e-mails are from verified domains.

If you have successfully finished with the initial setup (described in the **Prerequisites** section), continue with the procedure below.



## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *Google G Suite* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Specify the service URL:  <a href="https://www.googleapis.com/admin/directory">https://www.googleapis.com/admin/directory</a>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>  The authentication type in use is actually <b>OAuth</b> with JWT. But for any provisioning system based on OAuth, <b>BasicAuthentication</b> is used along with the <code>OAuth2TokenServiceURL</code> additional property.
User	Enter the service account's ID. You can take it from the <i>"client_email"</i> field in the JSON data, downloaded during the setup of Google service account.
Password	Enter the service account's private key, which represents a long string in PKCS8 format. You can take it from the <i>"private key"</i> field in the JSON data, downloaded during the setup of Google service account.
OAuth2TokenServiceURL	To make OAuth authentication to the Google G Suite system, enter the URL to the access token provider service. For more information, see <a href="#">Using OAuth 2.0 to Access Google APIs</a> .

Property Name	Description & Value
<code>jwt.subject</code>	<p>Enter the Google G Suite user on behalf of which the Google Directory API is called. This user has been assigned the role <b>User Management Admin</b>.</p> <p>This property corresponds to "sub" claim in JWT being generated during access token request: <a href="#">JWT: "sub" (Subject) Claim</a> ➔</p>
(Optional) <code>jwt.scope</code>	<p>Enter space-separated Google Directory API authorization scopes. For example:</p> <p><a href="https://www.googleapis.com/auth/admin.directory.user">https://www.googleapis.com/auth/admin.directory.user</a></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary Configuration:

```
ProxyType=Internet
Type=HTTP
Authentication=BasicAuthentication
URL=https://www.googleapis.com/admin/directory
User=1234567890-compute@developer.gserviceaccount.com
Password=-----BEGIN PRIVATE KEY-----\n123ABCDEFG123456789...
.../123456789ABCDEFG123=\n-----END PRIVATE KEY-----\n
OAuth2TokenServiceURL=https://www.googleapis.com/oauth2/v4/token
jwt.subject=john.smith@me123.accounts.ondemand.com
jwt.scope=https://www.googleapis.com/auth/admin.directory.user
```

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [Google G Suite](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

Transformation principles for the source system integration:

- **Mapping logic** – The provisioning framework reads all attributes from the Google G Suite source system and transfers them to the intermediate JSON data, which then tries to create consistent records in the target system, using all the available attributes accepted by the target system API. When a required attribute is missing, the default transformation is designed with a condition that will exclude the inconsistent records.
- **User off-boarding** – Identity Provisioning service is handling the deletion status of the users. When a user is deleted from Google G Suite, this deletion will be enforced into the target system as well.

You can change the default transformation mapping rules to reflect your current setup of entities in your Google G Suite. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Google Directory API: Users](#) ➡

[Google Directory API: Groups](#) ➡

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      },
      {
        "sourcePath": "$.primaryEmail",
        "targetPath": "$.emails[0].value",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.primaryEmail",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name"
      },
      {
        "constant": true,
        "targetPath": "$.active"
      },
      {
        "condition": "$.suspended == true",
        "constant": false,
        "targetPath": "$.active"
      }
    ]
  },
  "group": {
    "ignore": true,
    "mappings": [
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath": "$.schemas[0]"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.members[?(@.type == 'USER') && (@.status == 'ACTIVE')]",

```

```

        "targetPath": "$.members",
        "optional": true,
        "preserveArrayWithSingleElement": true
      },
      {
        "targetPath": "$.members[*].status",
        "type": "remove"
      },
      {
        "targetPath": "$.members[*].id",
        "type": "rename",
        "constant": "value"
      },
      {
        "constant": "display",
        "targetPath": "$.members[*].email",
        "type": "rename"
      },
      {
        "targetPath": "$.members[*].kind",
        "type": "remove"
      },
      {
        "targetPath": "$.members[*].etag",
        "type": "remove"
      },
      {
        "targetPath": "$.members[*].role",
        "type": "remove"
      }
    ]
  }
}

```

If the **displayName** attribute in the source system transformation does not provide group e-mails, you can modify the transformation the following ways:

- Map **email** to another attribute that contains a unique group e-mail.
- Concatenate the **displayName** attribute with your domain. For example:

#### Sample Code

```

{
  "sourcePath": "$.displayName",
  "targetPath": "$.email",
  "scope": "createEntity",
  "functions": [
    {
      "type": "concatString",
      "suffix": "@test.myaccount.ondemand.com"
    }
  ]
}

```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.37 LDAP Server

Follow this procedure to set up LDAP Server as a source system.

### Prerequisites

#### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#) or [Cloud Connector \(Cloud Foundry\)](#)
- **For tenants running on the infrastructure of SAP Cloud Identity Services:** You have a multi-environment subaccount in the Cloud Foundry region that maps the region of your Identity Authentication tenant and it is subscribed to the [Cloud Identity Services](#) application. For more information, see [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure](#).
- You have the credentials of a technical user in the LDAP Server, which is used to call the LDAP Server API to read the users and their attributes.

### Context

You can use LDAP Server to read entities from it and provision them to a target system. This scenario supports reading **users** and **groups**.

There are two versions of the LDAP Server connector. Both consume the LDAP Server API to read and write users and groups. The versions are handled by the `ldap.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on May 25, 2023) LDAP Server API version 1 is used. This is the default value of `ldap.api.version`.  
When using this version of the connector, the entities (users and groups) are read with all attributes.
- When the value is set to **2** – LDAP Server API version 2 is used.

This version of the connector comes with improved performance of the read operation for user and group attributes. You are now able to define which user and group attributes to be read. This is possible by adding values to the properties `ldap.user.attributes` or `ldap.group.attributes`.

Via these properties, you are able to add also user and group operational attributes (attributes which the directory organizes for internal use). For more information, refer to the official LDAP server documentation. After the additional values of the properties are set, the default read or proxy read transformations should also be adjusted accordingly.

For more information on how to update to LDAP Server connector version 2, see [Update Connector Version \[page 1484\]](#).

## Procedure

1. Add an access control system mapping for the **LDAP Server** in the Cloud Connector. This is needed to allow the Identity Provisioning service to access the LDAP server as a back-end system on the intranet. For more information, see [Configure Access Control \(LDAP\)](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add *LDAP Server* as a source system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>LDAP</i>
ldap.url	Specify the destination URL. It must be in the following format:  <code>ldap://&lt;external_host&gt;:&lt;external_port&gt;</code>
ldap.proxyType	Enter: <i>OnPremise</i>
ldap.authentication	Enter: <i>BasicAuthentication</i>

Property Name	Description & Value
<code>ldap.user</code>	Enter the <i>distinguishedName</i> of the technical LDAP user. This is the user you need to establish the connection and to perform all queries.
<code>ldap.password</code>	(Credential) Enter the password for the LDAP technical user.
<code>ldap.group.path</code>	Enter the complete path to the node containing the groups in the LDAP tree.
<code>ldap.user.path</code>	Enter the complete path to the users in the LDAP tree.
(Optional) <code>ldap.api.version</code>	<p>Defines the version of LDAP Server API.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <b>1</b> - Indicates that LDAP Server API version 1 is used.</li> <li>• <b>2</b> - Indicates that LDAP Server API version 2 is used.</li> </ul> <p>If the property is not defined - LDAP Server API version 1 is used.</p>
<code>ldap.user.attributes</code>	<p>Shows which user attributes from the source system to be included in the LDAP read result (and respectively, in the intermediate JSON data). Separate the attributes by comma (,).</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• LDAP Server version 1 - Even though you specified the attributes to be included in the LDAP read result, Identity Provisioning will return all attributes from the LDAP server.</li> <li>• LDAP Server version 2 - Only the specified attributes will be included in the LDAP server read result.</li> </ul>
<code>ldap.group.attributes</code>	<p>Shows which group attributes from the source system to be included in the LDAP read result (and respectively, in the intermediate JSON data). Separate the attributes by comma (,).</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• LDAP Server version 1 - Even though you specified the attributes to be included in the LDAP read result, Identity Provisioning will return all attributes from the LDAP server.</li> <li>• LDAP Server version 2 - Only the specified attributes will be included in the LDAP server read result.</li> </ul> <p>If nothing is set, all attributes are included.</p>

## → Remember

We strongly recommend that you enter different paths for LDAP users and groups. That means, the value of `ldap.user.path` should be different than the value of `ldap.group.path`.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

The LDAP Server source system is created by default with the properties listed below:

### Default LDAP Properties

```
ldap.user.attributes=
ldap.group.attributes=
ldap.user.object.class= inetOrgPerson
ldap.group.object.class= groupOfNames
ldap.group.uniqueName.attribute= cn
ldap.member.uniqueName.attribute= uid
ldap.attribute.group.id= cn
ldap.attribute.group.member= member
ldap.attribute.user.id= uid
ldap.attribute.dn= distinguishedName
ldap.user.filter=
ldap.group.filter=
ldap.page.size= 100
ldap.attribute.user.mail= mail
ldap.attribute.user.mobile= mobile
ldap.attribute.user.givenName= givenName
ldap.attribute.user.surname= sn
ldap.attribute.user.groups= memberOf
ldap.attribute.user.telephoneNumber= telephoneNumber
```

## i Note

The **ldap.attribute.\*** properties are used as parameterized properties in the default transformation. That is, if a property used in the transformation doesn't have a value, the provisioning job will fail when the transformation is loaded on runtime and the property value is substituted.

Also, you can change a property and use a new one (with a new name). In this case, you must replace the old property with the new one at all corresponding places in the transformation.



## 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *LDAP Server* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your LDAP server. For more information, see [Manage Transformations \[page 1494\]](#).

Currently, the default read transformations of the two connector versions have no differences, so there is no need to update them.

Before the read transformation, the LDAP Server attributes are represented as arrays (single-element arrays, or multi-value arrays separated by comma (,)). After read transformation (in the intermediate JSON data), the attributes are in SCIM format. For more information, see the official documentation for LDAP Server schema attributes in the **Related Information** section.

### i Note

When a user is deleted from LDAP Server, the deletion status is considered by the Identity Provisioning service during the read processes. Depending on the offboarding handling of the users in the target system, the user can be deleted, or can be set to *inactive*.

### Default transformation:

#### Code Syntax

```
{
  "user": {
    "mappings": [

      /* The value of entityIdSourceSystem is used to store the unique ID of the
      identity. You should not delete this statement.
      You could exchange the default attribute, resolved from
      ldap.attribute.user.id system property (which is used as a source) with
      another one but make sure the new source attribute is unique as well. */
      {
        "sourcePath": "$.%ldap.attribute.user.id%[0]",
        "targetVariable": "entityIdSourceSystem",
        "correlationAttribute": true
      },

      /* The value of the attribute resolved from ldap.attribute.user.id
      system property is used also as userName value for the internal JSON
      representation. */
      {
        "sourcePath": "$.%ldap.attribute.user.id%[0]",
        "targetPath": "$.userName",
        "correlationAttribute": true
      },

      /* The constant urn:ietf:params:scim:schemas:core:2.0:User is required as
      a value for the schemas definition in the Identity Authentication service
      SCIM REST API. */
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      }
    ]
  }
}
```

```

    },

    /* The value of the attribute resolved from ldap.attribute.user.mail
    system property is used also as a first array value in the emails JSON
    array. */
    {
        "sourcePath": "$.%ldap.attribute.user.mail%[0]",
        "targetPath": "$.emails[0].value",
        "optional": true,
        "correlationAttribute": true
    },

    /* The value of the attribute resolved from ldap.attribute.user.givenName
    system property is used for the name.givenName value in internal JSON
    representation. */
    {
        "sourcePath": "$.%ldap.attribute.user.givenName%[0]",
        "targetPath": "$.name.givenName",
        "optional": true
    },

    /* The value of the attribute resolved from ldap.attribute.user.surname
    system property is used for the name.familyName value in internal JSON
    representation. */
    {
        "sourcePath": "$.%ldap.attribute.user.surname%[0]",
        "targetPath": "$.name.familyName",
        "optional": true
    },

    /* The attribute resolved from ldap.attribute.user.groups system property
    is transformed by default into groups attribute of the SCIM internal
    representation: */
    {
        "sourcePath": "$.%ldap.attribute.user.groups%",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups[?(@.value)]"
    },
    {
        "sourcePath": "$.%ldap.attribute.user.mobile%[0]",
        "optional": true,
        "targetPath": "$.phoneNumbers[0].value"
    },
    {
        "condition": "$.%ldap.attribute.user.mobile%.length() > 0",
        "constant": "mobile",
        "targetPath": "$.phoneNumbers[0].type"
    },
    {
        "sourcePath": "$.%ldap.attribute.user.telephoneNumber%[0]",
        "optional": true,
        "targetPath": "$.phoneNumbers[1].value"
    },
    {
        "condition": "$.%ldap.attribute.user.telephoneNumber%.length()
> 0",
        "constant": "work",
        "targetPath": "$.phoneNumbers[1].type"
    }
]
},
"group": {
    "ignore": true,

```

```

    "mappings": [
      {
        "sourcePath": "$.%ldap.attribute.group.id%[0]",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.%ldap.attribute.group.id%[0]",
        "targetPath": "$.displayName"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath": "$.schemas[0]"
      },
      {
        "sourcePath": "$.%ldap.attribute.group.member%",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]"
      }
    ]
  }
}

```

As result of this mapping, this is how the data from LDAP Server looks like before and after the read transformation:

Transformation Snippet (from the user mapping)	Source JSON Data (as read from LDAP Server)	Intermediate JSON Data (as result from the transformation)
<pre> {   "user": {     "mappings": [       ...       {         "sourcePath": "\$.%ldap.attribute.u ser.groups%[0]",          "preserveArrayWithSi ngleElement": true,         "optional": true,         "targetPath": "\$ .groups[? (@.value)]"       },       ...     ]   } } </pre>	<pre> ... "memberOf": [   "SALES_US",   "SALES_EU",   "SALES_JA" ] ... </pre>	<pre> ... "groups": [   {     "value": "SALES_US"   },   {     "value": "SALES_EU"   },   {     "value": "SALES_JA"   } ] ... </pre>

## Note

By default, the **cn** attribute is returned for every read group. An administrator can change this behavior by setting the following properties:

- `ldap.group.uniquename.attribute` – the value can be either the CN or the whole DN (**distinguishedName**) of the group.

- `ldap.attribute.group.id` – the value can be CN or another attribute to be used as a group ID instead (for example, **displayName** or **description**).

For more information about these properties, see: [List of Properties \[page 94\]](#)

6. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Technical Documents](#) ➤

[Setting Timeout for Ldap Operations](#) ➤

[Connection Pooling Configuration](#) ➤

## 1.6.1.38 Microsoft Active Directory

Follow this procedure to set up Microsoft Active Directory as a source system.

## Prerequisites

### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)
- You have the credentials of a technical user in the Microsoft Active Directory, which is used to call the Microsoft Active Directory API to read the users and their attributes.

## Context

You can configure Microsoft Active Directory (Microsoft AD) as a source system to read users, groups and permission assignments.

There are two versions of the Microsoft AD connector. Both consume the LDAP Server API to read and write users and groups. The versions are handled by the `ldap.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on June 19, 2023) LDAP Server API version 1 is used. This is the default value of `ldap.api.version`.  
When using this version of the connector, the entities (users and groups) are read with all attributes. In this version, the `group members` attribute mapping in the proxy read transformation does not include `type` sub-attribute. In this case, all members are considered of type `user`, which is the sub-attribute fallback value. As a consequence, if the external system includes nested groups, they will not be handled properly.
- When the value is set to **2** – LDAP Server API version 2 is used.  
This version of the connector is with improved performance of the read operation for user and group attributes. You are now able to define which user and group attributes to be read. This is possible by adding values to the properties `ldap.user.attributes` or `ldap.group.attributes`.  
Via these properties, you are able to add also user and group operational attributes (attributes which the directory organizes for internal use). For more information, refer to the official LDAP server documentation. After the additional values of the properties are set, the default read or proxy read transformations should also be adjusted accordingly.  
In this version, the `group members` attribute mapping in the proxy read transformation is enhanced with `type` sub-attribute. The sub-attribute has two possible values – `user` and `group`. This allows you to read and preserve nested groups.  
For more information on how to update to Microsoft Active Directory version 2, see [Update Connector Version \[page 1484\]](#).

## Procedure

1. Add an access control system mapping for the **Microsoft Active Directory** in the Cloud Connector. This is needed to allow the Identity Provisioning service to access Microsoft AD as a back-end system on the intranet. For more information, see [Configure Access Control \(LDAP\)](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add **Microsoft Active Directory** as a source system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the **Properties** tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">LDAP</a>
ldap.url	Specify a destination URL. It must be in the following format:  ldap://<ext_host>:<ext_port>
ldap.proxyType	Enter: <a href="#">OnPremise</a>
ldap.authentication	Enter: <a href="#">BasicAuthentication</a>
ldap.attribute.user.id	Default property, which denotes the ID of a user.  By default, it's set to: <a href="#">cn</a>
ldap.attribute.group.id	Default property, which denotes the ID of a group.  By default, it's set to: <a href="#">cn</a>
ldap.attribute.dn	Default property, which denotes the distinguished name of a user or a group.  Only possible value: <a href="#">distinguishedName</a>
ldap.user	Enter the <a href="#">distinguishedName</a> or the <a href="#">userPrincipalName</a> of the Microsoft AD technical user. This is the user you need to establish the connection and to perform all queries.
ldap.password	(Credential) Enter the password for the Microsoft AD technical user.
ldap.group.path	Enter the complete path to the node containing the groups in Microsoft AD.
ldap.user.path	Enter the complete path to the users in Microsoft AD.
ldap.api.version	Defines the version of LDAP Server API.  <b>Possible values:</b> <ul style="list-style-type: none"><li>• <b>1</b> - Indicates that LDAP Server API version 1 is used.</li><li>• <b>2</b> - Indicates that LDAP Server API version 2 is used.</li></ul> If the property is not defined - LDAP Server API version 1 is used.

Property Name	Description & Value
<code>ldap.user.attributes</code>	<p>Shows which user attributes from the source system to be included in the Microsoft AD read result (and respectively, in the intermediate JSON data). Separate the attributes by comma (,).</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• Microsoft AD version 1 - Even though you specified the attributes to be included in the Microsoft AD read result, Identity Provisioning will return all attributes from the Microsoft AD.</li> <li>• Microsoft AD version 2 - Only the specified attributes will be included in the Microsoft AD read result. <a href="#">sAMAccountName</a> is always included as mandatory user attribute for Microsoft AD version 2.</li> </ul>
<code>ldap.group.attributes</code>	<p>Shows which group attributes from the source system to be included in the Microsoft AD read result (and respectively, in the intermediate JSON data). Separate the attributes by comma (,).</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• Microsoft AD version 1 - Even though you specified the attributes to be included in the Microsoft AD read result, Identity Provisioning will return all attributes from the Microsoft AD.</li> <li>• Microsoft AD version 2 - Only the specified attributes will be included in the or Microsoft AD.</li> </ul> <p>If nothing is set, all attributes are included.</p>

Property Name	Description & Value
(Optional) <code>ad.group.flatten</code>	<p>There are target systems that do not support nested groups (group structures). Therefore, if your Microsoft AD system contains such groups, they will not be resolved properly during the provisioning job. Such target systems are:</p> <ul style="list-style-type: none"> <li>• <a href="#">SAP Jam Collaboration</a></li> <li>• <a href="#">Identity Authentication</a></li> </ul> <p>To enable reading of group structures, you can use the <code>ad.group.flatten</code> property and set it to <a href="#">true</a>. It will read the group structure recursively and will "flatten" it so that all users from all groups and subgroups will be resolved and written in the target system as members of the main parent group.</p> <div> <p><b>i Note</b></p> <p>Performance might be affected when <code>ad.group.flatten</code> is set to true and group structure is very complex (contains a lot of nested groups).</p> </div> <p>For best results, we recommend you also set the system property <code>ldap.group.filter</code> whose value is one or multiple Microsoft AD parent groups. For example, if a parent group is named <b>Canteen</b>, the property should be set as: <code>ldap.group.filter=</code> <a href="#">(cn=Canteen)</a></p> <p>See the example below with multiple parent groups.</p>



Example for a destination or a set of properties:

```
Type=LDAP

Name=MyADDestination

ldap.user=john.smith@some.dummy.domain.com

ldap.password=*****

ldap.attribute.user.id=cn

ldap.attribute.group.id=cn

ldap.attribute.dn=distinguishedName

ldap.url=ldap://abcd:123

ldap.proxyType=OnPremise

ldap.authentication=BasicAuthentication

ldap.group.path=OU=Groups,OU=IAS,DC=global,DC=corp,DC=mycompany

ldap.user.path=OU=Users,OU=IAS,DC=global,DC=corp,DC=mycompany

ldap.group.filter=(|(cn=Canteen)(cn=Finance)(cn=Support_Team))

ad.group.flatten=true
```

---

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Microsoft Active Directory* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

Before the read transformation, the Microsoft Active Directory attributes are represented as arrays – single-element arrays, or multi-value arrays separated by comma (.). After the read transformation (in the intermediate JSON data), the attributes are in SCIM format. For more information about Microsoft AD schema attributes, see the **Related Information** section.

Currently, the default read transformations of the two connector versions have no differences, so there is no need to update them.

You can change the default transformation mapping rules to reflect your current setup of entities in Microsoft Active Directory. For more information, see:

[Manage Transformations \[page 1494\]](#)

[MS Graph: Users](#) ➡

[MS Graph: Groups](#) ➡

**Default transformation:**

## Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.ldap.attribute.user.id[0]",
        "targetVariable": "entityIdSourceSystem",
        "correlationAttribute": true
      },
      {
        "condition": "('%ldap.attribute.user.id' ==
'%ldap.attribute.dn%')",
        "constant": "",
        "functions": [
          {
            "function": "concatString",
            "prefix": "(?i)cn=.*?,(.*?)",
            "suffix": ",%ldap.user.path%"
          }
        ]
      },
      {
        "targetVariable": "nestedPathRegex"
      }
    ],
    // If a user is not a direct member of the configured user base path,
    // then function getMatchedRegexGroup extracts the nested path from the
    // distinguishedName of this user,
    // and maps the nested path to attribute
    [ 'urn:sap:cloud:scim:schemas:extension:ad:2.0:User' ][ 'nestedPath' ]
    {
      "condition": "('%ldap.attribute.user.id' ==
'%ldap.attribute.dn%')",
      "sourcePath": "$.ldap.attribute.user.id[0]",
      "functions": [
        {
          "function": "getMatchedRegexGroup",
          "regex": "${nestedPathRegex}",
          "groupIndex": 1
        }
      ]
    },
    "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:ad:2.0:User' ][ 'nestedPath' ]",
    "defaultValue": ""
  },
  {
    "sourcePath": "$.sAMAccountName[0]",
    "targetPath": "$.userName",
    "correlationAttribute": true
  },
  {
    "sourcePath": "$.displayName[0]",
    "targetPath": "$.displayName",
    "optional": true
  },
  {
    "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
    "targetPath": "$.schemas[0]"
  },
  {
    "sourcePath": "$.mail[0]",
    "targetPath": "$.emails[0].value",
    "optional": true,
    "correlationAttribute": true
  },
  {
  }
```

```

        "sourcePath": "$.givenName[0]",
        "targetPath": "$.name.givenName",
        "optional": true
    },
    {
        "sourcePath": "$.sn[0]",
        "targetPath": "$.name.familyName",
        "optional": true
    },
    {
        "sourcePath": "$.memberOf",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups[?(@.value)]"
    },
    {
        "sourcePath": "$.mobile[0]",
        "optional": true,
        "targetPath": "$.phoneNumbers[0].value"
    },
    {
        "condition": "$.mobile.length() > 0",
        "constant": "mobile",
        "targetPath": "$.phoneNumbers[0].type"
    },
    {
        "sourcePath": "$.telephoneNumber[0]",
        "optional": true,
        "targetPath": "$.phoneNumbers[1].value"
    },
    {
        "condition": "$.telephoneNumber.length() > 0",
        "constant": "work",
        "targetPath": "$.phoneNumbers[1].type"
    }
]
},
"group": {
    "ignore": true,
    "mappings": [
        {
            "sourcePath": "$.%ldap.attribute.group.id%[0]",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "condition": "('%ldap.attribute.group.id%' ==
'%ldap.attribute.dn%')",
            "constant": "",
            "functions": [
                {
                    "function": "concatString",
                    "prefix": "(?i)cn=.*?,(.*?)",
                    "suffix": ",%ldap.group.path%"
                }
            ],
            "targetVariable": "nestedPathRegex"
        }
    ],
    // If a group is not a direct member of the configured group base
    // path, then function getMatchedRegexGroup extracts the nested path from the
    // distinguishedName of this group,
    // and maps the nested path to attribute
    ['urn:sap:cloud:scim:schemas:extension:ad:2.0:Group']['nestedPath']
    {
        "condition": "('%ldap.attribute.group.id%' ==
'%ldap.attribute.dn%')",

```

```

        "sourcePath": "$.%ldap.attribute.group.id%[0]",
        "functions": [
            {
                "function": "getMatchedRegexGroup",
                "regex": "${nestedPathRegex}",
                "groupIndex": 1
            }
        ],
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:ad:2.0:Group']['nestedPath']",
        "defaultValue": ""
    },
    {
        "sourcePath": "$.sAMAccountName[0]",
        "targetPath": "$.displayName"
    },
    {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath": "$.schemas[0]"
    },
    {
        "sourcePath": "$.member",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.members[?(@.value)]",
        "optional": true
    }
]
}

```

As result of this mapping, that is how the data from Microsoft Active Directory looks like before and after the read transformation:

#### Source JSON Data

(as read from Microsoft Active Directory)

##### Sample Code

```

...
"memberOf": [
  "SALES_US",
  "SALES_EU"
]
...

```

#### Intermediate JSON Data

(as a result from the transformation)

##### Sample Code

```

...
"groups": [
  {
    "value": "SALES_US"
  },
  {
    "value": "SALES_EU"
  },
]
...

```

- Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).

2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Technical Documents](#) ➤

[Setting Timeout for Ldap Operations](#) ➤

[Connection Pooling Configuration](#) ➤

## 1.6.1.39 Microsoft Azure Active Directory

Follow this procedure to set up Microsoft Azure Active Directory (in short, Azure AD) as a source system.

### Prerequisites

- You've logged on to Microsoft Azure Portal, with credentials for a user with directory role **Global administrator**. For more information, see [Microsoft: Assigning administrator roles in Azure Active Directory](#) ➤.
- In ► [Azure Active Directory](#) ► [App registrations](#) ▾, you've registered an application with a secret key and permissions for Microsoft Graph API. These permissions must be consented by an administrator. For more information, see [Microsoft Graph permissions reference](#) ➤.
- (Relevant to target systems) Your registered application is assigned the **User Account Administrator** role. This role allows you to deprovision users. For more information, see [MS Azure PowerShell: Add-MsolRole Member](#) ➤.

#### i Note

If this role isn't assigned, you can only disable users. To do that, set the `accountEnabled` property to **false**. For more information, see [MS Graph: user resource type](#) ➤

### Permissions

Assign the following permissions to your application, according to your scenario. Also, the permissions have to be of type [Application](#).

- Users – [User.Read.All](#)
- Groups – [Group.Read.All](#)

For more information, see [MS Graph: Users](#) ➤ and [MS Graph: Groups](#) ➤

## Context

When using it as a source system, you can read both users and groups from Azure AD and provision them to any target system you've added in the Identity Provisioning user interface (if it supports groups).

If you've successfully finished with the initial setup (described in the **Prerequisites** section), continue with the procedure.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *Microsoft Azure Active Directory* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.



If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter: <b>https://graph.microsoft.com</b>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the application ID registered in your Azure AD subscription (see the <b>Prerequisites</b> section).
Password	Enter the secret key associated to your app registration.
aad.domain.name	Enter one of the verified domain names from the corresponding Azure AD tenant. On this domain, you perform the provisioning operations. For more information, see <a href="#">Microsoft Azure: Manage domain names</a> .

Property Name	Description & Value
<code>oauth.resource.name</code>	Enter: <a href="https://graph.microsoft.com">https://graph.microsoft.com</a>
<code>OAuth2TokenServiceURL</code>	Enter: <a href="https://login.microsoftonline.com/&lt;your_domain&gt;/oauth2/token">https://login.microsoftonline.com/&lt;your_domain&gt;/oauth2/token</a> , where <code>&lt;your_domain&gt;</code> is the domain name you have set in the <code>aad.domain.name</code> property.
(Optional) <code>aad.group.member.attributes</code>	<p>This property defines the attributes of a group member to be read by the Identity Provisioning. By default, it always reads the <b>type</b> and the <b>id</b> of a member.</p> <p>If you want the Identity Provisioning to read additional attributes, enter them as a single or a comma-separated value. For example:</p> <div> <p><b>❖ Example</b></p> <ul style="list-style-type: none"> <li>If you want to read the e-mails too, enter:  <code>aad.group.member.attributes=mail</code>  This reads a member's type, ID, and e-mail.</li> <li>If you want to read multiple additional attributes, enter:  <code>aad.group.member.attributes=mail,mobilePhone,displayName</code>  This reads a member's type, ID, e-mail, phone, and display name.</li> </ul> </div>
(Optional) <code>aad.user.attributes.membership.active</code>	<p>Use this property if you want to retrieve information about all the groups to which the users are assigned (if any).</p> <ul style="list-style-type: none"> <li>If the property is missing, or is set to <code>false</code> – group membership details for the users will not be extracted.</li> <li>If the property is set to <code>true</code> – group membership details for the users will be extracted.</li> </ul> <p>To learn more, see: <a href="#">List of Properties [page 94]</a></p>
(Optional) <code>aad.user.filter</code>	<p>Use this property to filter users by specific criteria, according to the <a href="#">Microsoft Graph REST API</a> .</p> <div> <p><b>i Note</b></p> <p>This property replaces the deprecated <code>msgraph-filter</code> property. To learn more, see: <a href="#">List of Properties [page 94]</a></p> </div>
(Optional) <code>aad.group.filter</code>	<p>Use this property to filter groups by specific criteria, according to the <a href="#">Microsoft Graph REST API</a> .</p>

Property Name	Description & Value
(Optional) <code>aad.user.filter.group.filter.combine</code>	<p>Use this property to filter users based on their group assignments.</p> <p>When set to <b>true</b>, this property combines user and group filters defined on the <code>aad.user.filter</code> and <code>aad.group.filter</code> properties to further narrow the search results. This way, only users that meet the following filtering criteria are returned:</p> <ul style="list-style-type: none"> <li>• Users that match the user filter and at the same time are members of groups that match the group filter.</li> <li>• Members of the filtered groups that match the user filter.</li> </ul> <p>When set to <b>false</b>, user and group filters are not combined.</p> <p>To learn more, see:</p> <p><a href="#">List of Properties [page 94]</a></p> <p><a href="#">Identity Provisioning: How to Get Users Based on Group Assignments from MS Azure AD</a> </p>
(Optional) <code>aad.user.attributes</code>	<p>Defines which user attributes are read from Microsoft Azure AD system.</p> <p>The property is set during system creation with the following default value:</p> <p><i><code>id,mail,userPrincipalName,displayName,mailNickname,givenName,surname,mobilePhone,businessPhones</code></i></p> <p>This means that by default, Identity Provisioning will read from MS Azure AD the user attributes defined in the property value. Those attributes are used in the default read transformation.</p> <p>To check the complete set of user attributes (properties) supported by Microsoft Azure AD, see: <a href="#">Microsoft Graph: User Properties</a> </p> <p>To learn more, see: <a href="#">List of Properties [page 94]</a></p>



Property Name	Description & Value
(Optional) <code>aad.group.attributes</code>	<p>Defines which group attributes are read from Microsoft Azure AD system.</p> <p>The property is set during system creation with the following default value: <code>id,displayName,mailNickname</code></p> <p>This means that by default, Identity Provisioning will read from MS Azure AD the group attributes defined in the property value and will also return the <code>members</code> attribute. Those attributes are used in the default read transformation.</p> <p>To check the complete set of group attributes (properties) supported by Microsoft Azure AD, see: <a href="#">Microsoft Graph: Group Properties</a> ➡</p> <p>To learn more, see: <a href="#">List of Properties [page 94]</a></p>
(Optional) <code>aad.entities.top</code>	<p>This property defines the number of entities to be read per page. Default value: <code>100</code></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Microsoft Azure Active Directory* source system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in your Microsoft Azure AD. For more information, see:

[Manage Transformations \[page 1494\]](#)

[MS Graph: Users](#) ➡

[MS Graph: Groups](#) ➡

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "$.userPrincipalName EMPTY false",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {

```

```

    "sourcePath": "$.mailNickname",
    "optional": true,
    "targetPath": "$.externalId",
    "correlationAttribute": true
  },
  {
    "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
    "targetPath": "$.schemas[0]"
  },
  {
    "sourcePath": "$.mail",
    "targetPath": "$.emails[0].value",
    "correlationAttribute": true
  },
  {
    "sourcePath": "$.userPrincipalName",
    "targetPath": "$.userName",
    "correlationAttribute": true
  },
  {
    "sourcePath": "$.displayName",
    "optional": true,
    "targetPath": "$.displayName"
  },
  {
    "sourcePath": "$.givenName",
    "optional": true,
    "targetPath": "$.name.givenName"
  },
  {
    "sourcePath": "$.surname",
    "optional": true,
    "targetPath": "$.name.familyName"
  },
  {
    "sourcePath": "$.mobilePhone",
    "optional": true,
    "targetPath": "$.phoneNumbers[0].value"
  },
  {
    "condition": "$.mobilePhone EMPTY false",
    "constant": "mobile",
    "targetPath": "$.phoneNumbers[0].type"
  },
  {
    "sourcePath": "$.businessPhones[0]",
    "optional": true,
    "targetPath": "$.phoneNumbers[1].value"
  },
  {
    "condition": "$.businessPhones.length() > 0",
    "constant": "work",
    "targetPath": "$.phoneNumbers[1].type"
  },
  {
    "sourcePath": "$.groups",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.groups"
  },
  {
    "sourcePath": "$.manager.id",
    "targetPath":
      "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
      ['value']",
    "optional": true
  }

```

```

    },
    {
      "sourcePath": "$.manager.displayName",
      "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
      "optional": true
    }
  ]
},
"group": {
  "ignore": true,
  "mappings": [
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": $.id,
      "targetVariable": "entityIdSourceSystem"
    },
    {
      "sourcePath": $.mailNickname,
      "optional": true,
      "targetPath": $.externalId
    },
    {
      "sourcePath": $.displayName,
      "targetPath": $.displayName
    },
    {
      "sourcePath": $.members,
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": $.members
    },
    {
      "targetPath": $.members[*].id,
      "constant": "value",
      "type": "rename",
      "optional": true
    }
  ]
}
}

```

## Custom Configurations

Goal	Action	Result
<p>You want Identity Provisioning to read the additional user attributes specified in property <code>aad.user.attributes</code> and write them successfully in the target system.</p>	<p>Extend the <i>"user"</i> mapping as follows:</p> <pre> {   "user": {     "condition": "\$userPrincipalName EMPTY false",     "mappings": [       {         "sourcePath": "\$",         "targetPath": "\$"       },       {         "sourcePath": "\$.id",         "targetVariable": "entityIdSourceSystem"       },       ...     ]   } } </pre>	<p>For example, you specify the <code>aad.user.attributes</code> property and set its value to:  <i><a href="#">id,mail,userPrincipalName,city,department,companyName</a></i></p> <p>As a result, every user in the target system will have the following attributes populated – ID, e-mail, user principle name, city, department, and company name.</p> <p>Returned information of an exemplary user:</p> <pre> ... {   "Resources": [     {       "id": "555-aaaa-333- abcd-111222333",       "mail": "john.smith@doma in.com",       "userPrincipalName": "ab c@something.onmicrosoft .com",       "city": "Sofia",       "department": "029",       "companyName": "SAP"     },     ...   ] } </pre>

Goal	Action	Result
<p>You want Identity Provisioning to read the additional group attributes specified in property <code>aad.group.attributes</code> and write them successfully in the target system.</p>	<p>Extend the "<i>group</i>" mapping as follows:</p> <pre> {   "group": {     "ignore": false,     "mappings": [       {         "sourcePath": "\$",         "targetPath": "\$"       },       {         "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",         "targetPath": "\$.\$schemas[0]"       },       ...     ]   } } </pre>	<p>Example: Specify the <code>aad.group.attributes</code> property and set its value to: <a href="#"><i>id,displayName,recommendation,isSubscribedByMail</i></a></p> <p>As a result, every group in the target system will have the following attributes populated – ID, display name, date and time of the last renewal, and information if it's subscribed by e-mail or not.</p> <p>Returned information of an exemplary group:</p> <pre> ... {   "Resources": [     {       "id": "12345-ccc-000-xyz-777888999",       "displayName": "ImportantGroup3",       "renewedDateTime": "2018-01-01T00:00:00Z",       "isSubscribedByMail": "true"     }   ] } ... </pre>

Goal	Action	Result
You want the returned value of a group member to be not the ID but a different attribute.	<p>In the "group" mapping, replace <b>id</b> with the new attribute. For example:</p> <p>If you replace <i>id</i> with <i>mail</i>, the transformation will look like this:</p> <pre>... {   "sourcePath":     "\$.members",   "preserveArrayWithSingleElement": true,   "optional": true,   "targetPath":     "\$.members" }, {   "targetPath":     "\$.members[*].mail",   "constant":     "value",   "type":     "rename",   "optional": true } ...</pre>	<p>Returned information of an exemplary group member:</p> <pre>... {   "members": [     {       "id": "5555555-aaaa-333- abcd-1111122223333",       "type": "user",       "value": "johnsmith@mail .acme.com"     }   ] } ...</pre>
	<p><b>⚠ Caution</b></p> <p>Make sure that you've added this attribute as a value of property <a href="#">aad.group.member.attributes</a>.</p>	

- Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.1.40 SCIM System

Follow this procedure to set up a SCIM system as a source system.

### Prerequisites

- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need this only if the SCIM system is exposed in a private corporate network. For more information, see [Cloud Connector](#).
- You have technical user credentials for a SCIM system, with read/write access permissions, depending on the scenario you want to implement. In case OAuth is used for authentication, client ID and secret are required when creating a destination for access token retrieval.

### Context

Create a general SCIM 2.0 based source system to read users and groups from it.

### Procedure

1. (Optional) If the SCIM system is exposed in a private corporate network, add an access control system mapping in Cloud Connector. For more information, see [Configure Access Control \(HTTP\)](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add *SCIM* as a source system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the *Properties* tab to configure the connection settings for your system.

#### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the service URL. For example:  <a href="#">http://&lt;cloudfoundry_server&gt;.com/api/uaa/</a>
ProxyType	Depending on your network exposure, enter one of the following: <ul style="list-style-type: none"> <li>• <a href="#">Internet</a></li> <li>• <a href="#">OnPremise</a></li> </ul>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	You can specify one of the following: <ul style="list-style-type: none"> <li>• Technical user ID</li> <li>• Client ID for OAuth HTTP destinations. It's used for retrieving of the access token.</li> </ul>
Password	You can enter one of the following: <ul style="list-style-type: none"> <li>• Technical user password</li> <li>• Client secret for OAuth HTTP destinations. It's used for retrieving of the access token.</li> </ul>
OAuth2TokenServiceURL	If you need to make OAuth authentication to the system, enter the URL to the access token provider service for OAuth HTTP destinations.  For example: <a href="#">https://&lt;token_provider&gt;.com/api/oauth2/v2.0/token</a>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SCIM](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SCIM system. For more information, see [Manage Transformations \[page 1494\]](#).

- **Mapping logic** – the behavior of the default transformation logic is to read all user attributes from the source SCIM system, and then map them to the internal SCIM representation. It uses `entityIdSourceSystem` to store the unique ID of the identity. The ID is removed by default, because it is specific for the source system.
- **User off-boarding** – it depends on the target system API. When a user is deleted from the SCIM system, the deletion status is considered and depends on the user status handling of the target system. The user will be either deleted or set as **inactive**.



## Default transformation:

### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "targetPath": "$",
        "sourcePath": "$"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "sourcePath": "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "type": "remove",
        "targetPath": "$.id"
      },
      {
        "type": "remove",
        "targetPath": "$.meta"
      }
    ]
  },
  "group": {
    "ignore": true,
    "mappings": [
      {
        "targetPath": "$",
        "sourcePath": "$"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      },
      {
        "type": "remove",
        "targetPath": "$.id"
      },
      {
        "type": "remove",
        "targetPath": "$.meta"
      }
    ]
  }
}
```

6. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

### 1.6.1.41 SSH Server (Beta)

Follow this procedure to set up an SSH server (Beta) as a source system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- You have credentials for a tenant in SAP Business Technology Platform. For more information, see: [Accounts](#)
- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need this only when your SSH server resides in a remote system, outside your Neo environment. For more information, see [Cloud Connector](#).

### i Note

This is a beta feature available on SAP Business Technology Platform. For more information, see: *Enable beta features* in [Change Subaccount Details](#)

## Context

**SSH Server** is a system (connector) in beta state. It helps you execute bash scripts through SSH connection. The configuration allows you to attach separate scripts per entity lifecycle callback (such as user create, group create/update, and so on). This system helps you connect to remote machines via SSH tunnel, with or without use of the Cloud Connector, depending on whether the SSH port is visible or not.

The bash scripts can take as parameters fields that are coming from the entity JSON data. For example: `sudo su - vcap /home/myscript.sh $.userName $.email`

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SSH Server (Beta)* as a source system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

Below are listed all available SSH Server properties. Some of them can be mandatory and others – optional, depending on your scenario.

Mandatory Properties

Property Name	Description & Value
ProxyType	Possible values: <ul style="list-style-type: none"><li>• <b>Internet</b> – if the SSH port is visible in your Neo environment</li><li>• <b>OnPremise</b> – if the SSH port is not directly accessible, and you have to use the Cloud Connector. You have to configure TCP protocol connection to the SSH host and port (specify the configuration properties <code>ssh.host</code> and <code>ssh.port</code>).</li></ul>
CloudConnectorLocationId	Relevant when the proxy type is <i>OnPremise</i> . Use it only if your SAP Business Technology Platform account uses more than one Cloud Connector.
ssh.auth.type	Supported SSH authentication types: <ul style="list-style-type: none"><li>• <b>key</b></li><li>• <b>pwd</b></li><li>• <b>otp</b></li><li>• <b>key+otp</b></li><li>• <b>key+pwd</b></li><li>• <b>pwd+otp</b></li><li>• <b>key+pwd+otp</b></li></ul>
ssh.host	

Property Name	Description & Value
<code>ssh.port</code>	22
<code>ssh.username</code>	
<code>ssh.password</code>	<p>(Credential) Taken into account only if the authentication type includes <b>pwd</b>. That means any of the following:</p> <ul style="list-style-type: none"> <li><code>ssh.auth.type = <i>pwd</i></code></li> <li><code>ssh.auth.type = <i>pwd+otp</i></code></li> <li><code>ssh.auth.type = <i>key+pwd</i></code></li> <li><code>ssh.auth.type = <i>key+pwd+otp</i></code></li> </ul>
<code>ssh.totp.secret.key</code>	<p>(Credential) Taken into account only if the authentication type includes <b>otp</b>. That means any of the following:</p> <ul style="list-style-type: none"> <li><code>ssh.auth.type = <i>otp</i></code></li> <li><code>ssh.auth.type = <i>key+otp</i></code></li> <li><code>ssh.auth.type = <i>pwd+otp</i></code></li> <li><code>ssh.auth.type = <i>key+pwd+otp</i></code></li> </ul>
<code>ssh.private.key.type</code>	<p>The type of the SSH private key. Possible values:</p> <ul style="list-style-type: none"> <li><b>ssh-rsa</b></li> <li><b>ssh-dsa</b></li> </ul> <p>Default value: <i>ssh-rsa</i></p> <div> <p><b>i Note</b></p> <p>If you choose <i>ssh-rsa</i>, the key should be in format <b>PKCS #8</b>, non-encrypted.</p> </div>
<code>ssh.private.key</code>	<p>(Credential) Taken into account only if the authentication type includes <b>key</b>. That means any of the following:</p> <ul style="list-style-type: none"> <li><code>ssh.auth.type = <i>key</i></code></li> <li><code>ssh.auth.type = <i>key+pwd</i></code></li> <li><code>ssh.auth.type = <i>key+otp</i></code></li> <li><code>ssh.auth.type = <i>key+pwd+otp</i></code></li> </ul>
<code>ssh.read.groups.command</code>	Path to the bash command you need to execute to read groups.
<code>ssh.read.users.command</code>	Path to the bash command you need to execute to read users.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a

default transformation for the [SSH Server \(Beta\)](#) source system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SSH server. For more information, see [Manage Transformations \[page 1494\]](#).

**Default transformation:**

**Code Syntax**

```
{
  "user": {
    "mappings": [
      {
        "targetPath": "$",
        "sourcePath": "$"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      }
    ]
  },
  "group": {
    "ignore": true,
    "mappings": [
      {
        "targetPath": "$",
        "sourcePath": "$"
      },
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem"
      }
    ]
  }
}
```

5. Now, add a target system to provision users and groups into it. Choose from: [Target Systems \[page 702\]](#)

## Next Steps

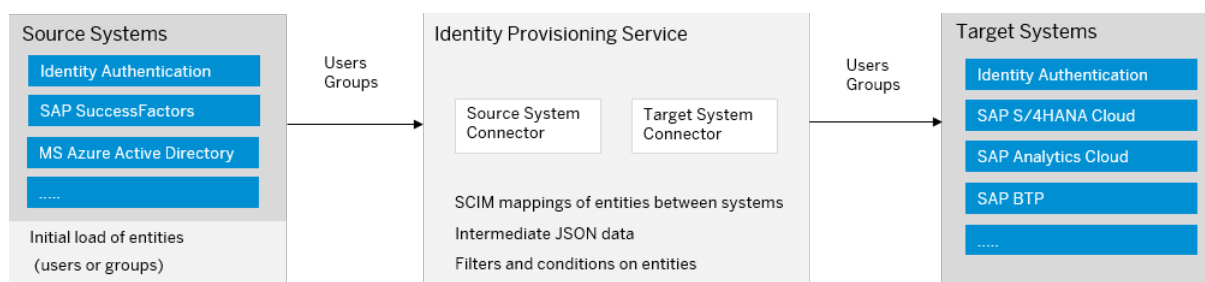
1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2 Target Systems

A target system is the connector used for writing entities.

Target systems are usually cloud systems, where the Identity Provisioning service creates or updates the entities read from the source system.

A target system can be connected to a single or multiple source systems. In the case of multiple source systems, we recommend that you run the provisioning jobs **successively** for each system, not simultaneously. This way, you'll avoid incorrect overwriting or merging of entity data, hence failed provisioning jobs.



### ! Restriction

By default, the maximum number of productive target systems you are allowed to add for your tenant is **50**.

If your business requires using more systems, create an incident for component [BC-IAM-IPS](#) to request them. Describe your scenarios and provide a reason why you need the additional systems.

## Related Information

[System Types \[page 86\]](#)

### 1.6.2.1 Identity Authentication

Follow this procedure to set up SAP Cloud Identity Service – Identity Authentication as a target system.

## Prerequisites

To establish the connection between Identity Provisioning and Identity Authentication, you need to set up the technical user (of type [System](#)) in Identity Authentication and assign this user the necessary authorizations. You can do it now (as a prerequisite) or in the process of configuring Identity Authentication as a target system, as described in step 3.

## Context

Identity Authentication provides authentication and single sign-on for users in the cloud.

Using Identity Provisioning, you can read corporate users from on-premise or cloud systems, and provision them to the Identity Authentication user store. This way, you can implement secure authentication, single sign-on (SSO), strong authentication and mobile SSO so that the provisioned users have access to the business applications of your company. For example, you can implement two-factor authentication and mobile SSO for SAP SuccessFactors users.

You can use the Identity Provisioning UI to configure Identity Authentication as a target system where you can provision users, groups and group members. The target system consumes SCIM 2.0 API provided by Identity Authentication.

There are two versions of the Identity Authentication SCIM API. They are controlled by the `ias.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on July 9, 2021) - Identity Authentication SCIM API (in short, SCIM API version 1) is used. For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#)
- When the value is set to **2** - Identity Directory SCIM API (in short, SCIM API version 2) is used. This value is set automatically for all manually created systems in the Identity Provisioning UI after versioning was introduced on July 9, 2021.

SCIM API version 2 does not support managing of group assignments via the SCIM user resource. The "groups" attribute of the user is read-only. This means that the user group assignments should be managed via the SCIM group resources using the "members" attribute (as it is defined by the SCIM standard). Also, the group resource mapping in the transformation is not ignored by default, as it is in SCIM API version 1.

### i Note

Identity Authentication (using SCIM API version 2) and Identity Directory are sometimes used interchangeably. Identity Directory is the persistency layer of SAP Cloud Identity Services – Identity Authentication.

To create Identity Authentication as a target system, proceed as follows:

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *Identity Authentication* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and Identity Authentication and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added Identity Authentication target system, select the [Certificate](#) tab and choose [Generate](#) > [Download](#) as described in [Manage Certificates \[page 1506\]](#).

Skip step **a.** if you want to use basic authentication.

In SAP Cloud Identity Services administration console, perform the next steps. They are relevant for both basic and certificate-based authentication.

- b. [Add System as administrator](#) and provide the respective credentials.

For basic authentication, provide a password. The user ID will be generated automatically when you set the password for the first time.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. Save your changes.
- d. Make sure [Manage Users](#) and [Manage Groups](#) authorization roles are enabled for the technical user. This way, you can create, edit and delete users and groups in the Identity Authentication user store.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL of the Identity Authentication tenant of your company.</p> <p>For example: <b>https://mytenant.accounts.on-demand.com</b></p> <div> <h3>i Note</h3> <p>If your Identity Authentication Shanghai (China) tenants reside on SAP BTP, Neo environment, you should use the following URL pattern: <a href="#">https://&lt;tenant_ID&gt;.accounts.sapcloud.cn/</a></p> </div>
ProxyType	<p>Enter: <a href="#">Internet</a></p> <p>The Identity Authentication is a cloud solution and is outside of your company on-premise infrastructure.</p>



Property Name	Description & Value
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"> <li>• <a href="#">BasicAuthentication</a></li> <li>• <a href="#">ClientCertificateAuthentication</a></li> </ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the Client ID (previously User ID) of the Identity Authentication technical user. It is generated automatically for the administrator of type system, when choosing <a href="#">Secrets</a> &gt; <a href="#">Add</a> &gt; <a href="#">Save</a> . For example: <a href="#">1ab7c243-5de5-4530-8g14-1234h26373ab</a></p> <p>If your technical user was created before <i>January 2020</i>, enter the T-user. For example: <a href="#">T000003</a></p>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the Client Secret (previously Password) of the Identity Authentication technical user. It is generated automatically for the administrator of type system, when choosing <a href="#">Secrets</a> &gt; <a href="#">Add</a> &gt; <a href="#">Save</a> .</p>
Optional Properties	
Property Name	Description & Value
<ul style="list-style-type: none"> <li>• <code>ias.&lt;property_name&gt;</code></li> <li>• <code>scim.&lt;property_name&gt;</code></li> </ul>	<p>When using SCIM API version 2, property names start with <code>ias</code> prefix, for example: <code>ias.user.unique.attribute</code>.</p> <p>When using SCIM API version 1, property names start with <code>scim</code> prefix, for example: <code>scim.user.unique.attribute</code>.</p> <p>For more information, see <a href="#">List of Properties [page 94]</a>. Use the main search or filter properties by <a href="#">Name</a> or <a href="#">System Type</a> columns.</p>

Property Name	Description & Value
<code>ias.user.unique.attribute</code>	<p>This property defines by which unique attribute(s) an existing user will be resolved in the event of conflicting users. It appears by default when the system is created, and its value is set to <code>userName</code>.</p> <p>Other possible values:</p> <ul style="list-style-type: none"> <li>• <code>emails[0].value</code></li> <li>• <code>userName,emails[0].value</code></li> <li>• <code>phoneNumbers[0].value</code></li> <li>• <code>externalId</code>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>If the service finds a user on the target system through this filter, then the conflicting user will overwrite the existing one. If the service does not find a user on the target system through this filter, the creation will fail.</p> <p>For more information, see: <a href="#">List of Properties [page 94]</a></p>
<code>ips.failed.request.retry.attempts</code>	Predefined value: <code>2</code>
<code>ips.failed.request.retry.attempts.interval</code>	Predefined value: <code>60</code>

##### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Identity Authentication* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

When Identity Authentication is configured as a target system, the default transformation logic writes all the user attributes in the Identity Authentication user store. The logic is provided by the Identity Authentication SCIM REST API, which then maps the attributes to the internal SCIM representation.

You can change the default transformation mapping rules to reflect your current setup of entities in your Identity Authentication system. For more information, see: [Manage Transformations \[page 1494\]](#).

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

SCIM API version 1: [Identity Authentication: SCIM REST API](#)

SCIM API version 2: [Identity Directory SCIM API](#) 

When Identity Authentication is configured as a target system, the default transformation logic:

- Reads all user attributes from the intermediate SCIM representation.
- Skips some of the attributes from the identity records.

This way, the transformation logic ensures that the identity data, sent to the Identity Provisioning SCIM REST API, is consistent.

##### Default transformation for SCIM API version 1:

## Code Syntax

```
{
  "user": {
    "condition": "($.emails.length() > 0) && ($.name.familyName EMPTY
false) && isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "sourcePath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.corporateGroups"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": "$.schemas[1]"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "optional": true
      },
      {
        "sourcePath": "$.emails[*].value",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails[?(@.value)]"
      },
      {
        "sourcePath": "$.userType",
        "targetPath": "$.userType",
        "optional": true
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.name.honorificPrefix",
        "targetPath": "$.name.honorificPrefix",
        "optional": true
      },
      {
        "sourcePath": "$.addresses",
        "targetPath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "defaultValue": [],
        "optional": true,
      }
    ]
  }
}
```

```

        "functions": [
            {
                "function": "putIfAbsent",
                "key": "type",
                "defaultValue": "work"
            },
            {
                "condition": "(@.type NIN ['work', 'home'])",
                "function": "putIfPresent",
                "key": "type",
                "defaultValue": "work"
            }
        ]
    },
    {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
    },
    {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
    },
    {
        "sourcePath":
        "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
        ['employeeNumber']",
        "targetPath":
        "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
        ['employeeNumber']",
        "optional": true
    },
    {
        "sourcePath":
        "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
        ['costCenter']",
        "targetPath":
        "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
        ['costCenter']",
        "optional": true
    },
    {
        "sourcePath":
        "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
        ['organization']",
        "targetPath":
        "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
        ['organization']",
        "optional": true
    },
    {
        "sourcePath":
        "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
        ['division']",
        "targetPath":
        "$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
        ['division']",
        "optional": true
    },
    {

```

```

        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['manager']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['manager']
['value']",
        "optional" : true,
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['manager']
['displayName']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['manager']
['displayName']",
        "optional" : true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "defaultValue": true,
        "optional": true
    },
    {
        "constant": "false",
        "targetPath": "$.sendMail",
        "scope": "createEntity"
    },
    {
        "constant": "true",
        "targetPath": "$.mailVerified",
        "scope": "createEntity"
    },
    {
        "constant": "disabled",
        "targetPath": "$.passwordStatus",
        "scope": "createEntity"
    },
    {
        "constant": "39",
        "targetPath": "$.sourceSystem",
        "scope": "createEntity"
    },
    {
        "constant": "employee",
        "targetPath": "$.userType"
    },
    {
        "sourcePath": "$.timezone",
        "targetPath": "$.timeZone",
        "optional": true
    }
]
},

```

```

"group": {
  "ignore": true,
  "mappings": [
    {
      "sourceVariable": "entityIdTargetSystem",
      "targetPath": "$.id"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName"
    },
    {
      "scope": "createEntity",
      "sourcePath": "$.displayName",
      "targetPath": "$"
    }
  ],
  "functions": [
    {
      "type": "replaceAllString",
      "regex": "[\\s\\p{Punct}]",
      "replacement": "_"
    }
  ]
},
{
  "scope": "createEntity",
  "optional": true,
  "sourcePath": "$"
},
{
  "targetPath": "$"
},
{
  "optional": true,
  "sourcePath": "$"
},
{
  "targetPath": "$"
},
{
  "optional": true,
  "preserveArrayWithSingleElement": true,
  "sourcePath": "$.members[*].value",
  "targetPath": "$.members[?(@.value)]",
  "functions": [
    {
      "type": "resolveEntityIds"
    }
  ]
}
]
}
}

```

#### Default transformation for SCIM API version 2:

##### Code Syntax

```

{
  "user": {
    "condition": "($.emails EMPTY false) && ($.userName EMPTY false) &&
isValidEmail($.emails[0].value)",
    "mappings": [
      {

```

```

        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
    },
    {
        "constant":
        [ "urn:ietf:params:scim:schemas:core:2.0:User", "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User", "urn:ietf:params:scim:schemas:extension:sap:2.0:User", "urn:sap:cloud:scim:schemas:extension:custom:2.0:User" ],
        "targetPath": "$.schemas"
    },
    {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
    },
    {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$
        [ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ] [ 'emails' ]",
        "scope": "createEntity",
        "functions": [
            {
                "function": "putIfAbsent",
                "key": "verified",
                "defaultValue": true
            }
        ]
    },
    {
        "targetPath": "$[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
        [ 'emails' ] [ * ] [ 'type' ]",
        "type": "remove"
    },
    {
        "sourcePath": "$.emails[ * ].value",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails[ ? ( @.value ) ]"
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
    },
    {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true
    },
    {
        "sourcePath": "$.name.honorificPrefix",
        "targetPath": "$.name.honorificPrefix",
        "optional": true
    },
    {
        "sourcePath": "$.addresses",
        "targetPath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "defaultValue": [],
        "optional": true,
        "functions": [
            {
                "function": "putIfAbsent",

```

```

        "key": "type",
        "defaultValue": "work"
      },
      {
        "condition": "(@.type NIN ['work', 'home'])",
        "function": "putIfPresent",
        "key": "type",
        "defaultValue": "work"
      }
    ]
  },
  {
    "sourcePath": "$.phoneNumbers",
    "targetPath": "$.phoneNumbers",
    "preserveArrayWithSingleElement": true,
    "optional": true
  },
  {
    "sourcePath": "$.displayName",
    "targetPath": "$.displayName",
    "optional": true
  },
  {
    "sourcePath": "$.userType",
    "targetPath": "$.userType",
    "optional": true
  },
  {
    "sourcePath": "$.locale",
    "targetPath": "$.locale",
    "optional": true
  },
  {
    "sourcePath": "$.timezone",
    "targetPath": "$.timezone",
    "optional": true
  },
  {
    "sourcePath": "$.active",
    "targetPath": "$.active",
    "defaultValue": true,
    "optional": true
  },
  {
    "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
    "optional": true
  },
  {
    "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
    "optional": true
  },
  {
    "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
    "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
    "optional": true
  },
  {

```



```

        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['value']",
        "optional" : true,
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['value']",
        "optional" : true
    },
    {
        "constant": false,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['sendMail']",
        "scope": "createEntity"
    },
    {
        "constant": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['mailVerified']",

```

```

        "scope": "createEntity"
      },
      {
        "constant": "disabled",
        "targetPath": "$['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
        ['passwordDetails']['status']",
        "scope": "createEntity"
      },
      {
        "sourcePath": "$
        ['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes']",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$
        ['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes']"
      },
      {
        "constant": "<your-initial-password>",
        "targetPath": "$.password",
        "scope": "createEntity",
        "ignore": true
      },
      {
        "constant": "<your-source-system-type-code>",
        "targetPath": "$
        ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystem']",
        "scope": "createEntity",
        "ignore": true
      },
      {
        "constant": "<your-source-system-id>",
        "targetPath": "$
        ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId']",
        "scope": "createEntity",
        "ignore": true
      }
    ]
  },
  "group": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant":
        ["urn:ietf:params:scim:schemas:core:2.0:Group", "urn:sap:cloud:scim:schemas:
        extension:custom:2.0:Group"],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]",
        "functions": [
          {
            "entityType": "user",
            "type": "resolveEntityIds"
          }
        ]
      }
    ]
  },
}

```

```

        "sourcePath": "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]",
        "functions": [
            {
                "entityType": "group",
                "type": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath": "$.displayName",
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['description']",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['description']"
    }
]
}
}

```

### → Remember

If you set `$.userType` to **"public"**, all passwords will be written by default in the Identity Authentication. Thus, all provisioned users will successfully sign in to Identity Authentication target system.


When `$.userType` is set to **"employee"**, the sign-in behavior of the provisioned users depends on whether users have been created with or without a password, and where these passwords are stored. Thus, you need to modify the target transformations accordingly in order for the users to successfully sign in to the Identity Authentication console.


To learn more, see [Guided Answers: Identity Authentication: Provisioned Users Can't Sign In](#) 

6. Add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

### ⚠ Caution

By default, the **email** attribute is unique for *Identity Authentication* but it might not be unique for other systems, such as *SAP SuccessFactors*. That's why, when you choose a source system, make sure that there are no users with duplicate e-mails. If there are such, then after the provisioning job all users with the same e-mail address will be created as a single user in *Identity Authentication*.

To learn more about how to fix this issue or prevent it from happening, see [Guided Answers: Multiple Users from a Source System Are Created as One in the Target](#) 

To learn more about excluding users from Identity Authentication target, see: [Exclude Users from Provisioning to Target Systems](#) 

If you have configured the **email** attribute to NOT be unique in *Identity Authentication*, then ignore this **Caution** note and the guided answer. For more information about configurable attributes, see: [Identity Authentication: Configure User Identifier Attributes](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Identity Authentication: Documentation](#)

[Identity Authentication: SCIM REST API](#)

(Guided Answers) [Multiple Users from a Source System Are Created as One in the Target](#) 

(Guided Answers) [Identity Authentication: Provisioned Users Can't Sign In](#) 

## 1.6.2.2 Local Identity Directory

Follow this procedure to set up Local Identity Directory as a target system.

## Prerequisites

### Note

The *Local Identity Directory* connector is available for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

## Context

Identity directory is the user store of SAP Cloud Identity Services. It provides a central place for storing and managing users, groups and custom schemas through the System for Cross-domain Identity Management 2.0 REST API, in short Identity Directory SCIM API.

You can provision users and groups to the identity directory from various source systems by configuring the Local Identity Directory connector as a target system in the SAP Cloud Identity Services administration console. The entities are written to the identity directory of your current SAP Cloud Identity Services tenant. Therefore, you don't have to set up connectivity and authentication properties for the target system. In case you want to write entities to the identity directory in another tenant, add Identity Authentication (version 2) as a target system and configure the respective connectivity and authentication properties.

To create Local Identity Directory as a target system, proceed as follows:

## Procedure

1. Sign in to SAP Cloud Identity Services administration console and navigate to ► [Identity Provisioning](#) ► [Target Systems](#) .

2. Add [Local Identity Directory](#) as a target system.

For more information, see [Add a System \[page 1477\]](#).

3. **Optional:** Configure properties. For example, bulk properties to enable bulk operations:  
`ids.bulk.operations.max.count`.


Local Identity Directory properties are prefixed with `ids.<property_name>`. For more information, see [List of Properties \[page 94\]](#).

4. **Optional:** Configure transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the Local Identity Directory target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

When Local Identity Directory is configured as a target system, the default transformation logic writes the user attributes in the user store of SAP Cloud Identity Services. The logic is provided by the Identity Directory SCIM API, which then maps the attributes to the internal SCIM representation.

You can change the default transformation mapping rules to reflect your current setup of entities in your Local Identity Directory. For more information, see:

- [Manage Transformations \[page 1494\]](#)
- SCIM API version 2: [Identity Directory SCIM API](#) 

### Default transformation for Local Identity Directory:

#### Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY false) && ($.userName EMPTY false)",
    "mappings": [
```

```

    {
      "sourceVariable": "entityIdTargetSystem",
      "targetPath": "$.id"
    },
    {
      "constant":
[ "urn:ietf:params:scim:schemas:core:2.0:User", "urn:ietf:params:scim:schemas
:extension:enterprise:2.0:User", "urn:ietf:params:scim:schemas:extension:sap
:2.0:User", "urn:sap:cloud:scim:schemas:extension:custom:2.0:User" ],
      "targetPath": "$.schemas"
    },
    {
      "sourcePath": "$.userName",
      "targetPath": "$.userName"
    },
    {
      "constant": "userName",
      "targetVariable": "entityCorrelationAttributeName"
    },
    {
      "sourcePath": "$.userName",
      "targetVariable": "entityCorrelationAttributeValue"
    },
    {
      "sourcePath": "$.emails",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]['emails']",
      "scope": "createEntity",
      "functions": [
        {
          "function": "putIfAbsent",
          "key": "verified",
          "defaultValue": true
        }
      ]
    },
    {
      "targetPath": "$[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
['emails'][*]['type']",
      "type": "remove"
    },
    {
      "sourcePath": "$.emails[*].value",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.emails[?(@.value)]"
    },
    {
      "sourcePath": "$.name.givenName",
      "targetPath": "$.name.givenName",
      "optional": true
    },
    {
      "sourcePath": "$.name.middleName",
      "targetPath": "$.name.middleName",
      "optional": true
    },
    {
      "sourcePath": "$.name.familyName",
      "targetPath": "$.name.familyName",
      "optional": true
    },
    {
      "sourcePath": "$.name.honorificPrefix",
      "targetPath": "$.name.honorificPrefix",
      "optional": true
    }
  ],

```

```

    {
      "sourcePath": "$.addresses",
      "targetPath": "$.addresses",
      "preserveArrayWithSingleElement": true,
      "defaultValue": [],
      "optional": true,
      "functions": [
        {
          "function": "putIfAbsent",
          "key": "type",
          "defaultValue": "work"
        },
        {
          "condition": "(@.type NIN ['work', 'home'])",
          "function": "putIfPresent",
          "key": "type",
          "defaultValue": "work"
        }
      ]
    },
    {
      "sourcePath": "$.phoneNumbers",
      "targetPath": "$.phoneNumbers",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "optional": true
    },
    {
      "sourcePath": "$.userType",
      "targetPath": "$.userType",
      "optional": true
    },
    {
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "optional": true
    },
    {
      "sourcePath": "$.timezone",
      "targetPath": "$.timezone",
      "optional": true
    },
    {
      "sourcePath": "$.active",
      "targetPath": "$.active",
      "defaultValue": true,
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
      "optional": true
    },
    {

```

```

        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['value']",
        "optional" : true,
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['displayName']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['displayName']",
        "optional" : true
    },
    {

```



```

        "constant": false,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sendMail']",
        "scope": "createEntity"
    },
    {
        "constant": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['mailVerified']",
        "scope": "createEntity"
    },
    {
        "constant": "disabled",

"targetPath": "$['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['passwordDetails']['status']",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$
    ['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes']",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$
    ['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes']"
    },
    {
        "constant": "<your-initial-password>",
        "targetPath": "$.password",
        "scope": "createEntity",
        "ignore": true
    },
    {
        "constant": "<your-source-system-type-code>",
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystem']",
        "scope": "createEntity",
        "ignore": true
    },
    {
        "constant": "<your-source-system-id>",
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId']",
        "scope": "createEntity",
        "ignore": true
    }
]
},
"group": {
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "constant":
    ["urn:ietf:params:scim:schemas:core:2.0:Group","urn:sap:cloud:scim:schemas:
extension:custom:2.0:Group"],
            "targetPath": "$.schemas"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName"
        },
        {
            "sourcePath": "$.members[*].value",
            "preserveArrayWithSingleElement": true,
            "optional": true,

```

```

        "targetPath": "$.members[?(@.value)]",
        "functions": [
            {
                "entityType": "user",
                "type": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath": "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]",
        "functions": [
            {
                "entityType": "group",
                "type": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath": "$.displayName",
        "targetPath": "$",
        [ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$",
        [ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
        "optional": true,
        "targetPath": "$",
        [ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$",
        [ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'description' ]",
        "optional": true,
        "targetPath": "$",
        [ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'description' ]"
    }
]
}

```

5. Add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

### ⚠ Caution

By default, the **email** attribute is unique for Local Identity Directory but it might not be unique for other systems, such as [SAP SuccessFactors](#). That's why, when you choose a source system, make sure that there are no users with duplicate e-mails. If there are such, then after the provisioning job all users with the same e-mail address will be created as a single user in Local Identity Directory.

To learn more about how to fix this issue or prevent it from happening, see [Guided Answers: Multiple Users from a Source System Are Created as One in the Target](#) 📖

To learn more about excluding users from Local Identity Directory target, see: [Exclude Users from Provisioning to Target Systems](#) 📖

If you have configured the **email** attribute to NOT be unique in Local Identity Directory, then ignore this **Caution** note and the guided answer. For more information about configurable attributes, see: [Configure User Identifier Attributes](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

(Guided Answers) [Multiple Users from a Source System Are Created as One in the Target](#) 📖

(Guided Answers) [Identity Authentication: Provisioned Users Can't Sign In](#) 📖

## 1.6.2.3 SAP Advanced Financial Closing

Follow this procedure to set up SAP Advanced Financial Closing as a target system.

### Prerequisites

You have created an instance and generated a service key for the standard service plan of SAP Advanced Financial Closing. For more information, see: [How to Manage User Access Using the SCIM API Provided](#).

The service key contains the API URL and the OAuth credentials (`clientid` and `clientsecret`) under the `uaa` property.

### Context

SAP Advanced Financial Closing allows you to define, automate, process, and monitor the financial closing tasks for the entities of your organization. It is an SAP BTP application that runs in an SAP BTP subaccount.

You can use Identity Provisioning to configure SAP Advanced Financial Closing as a target system for provisioning users, user groups and user roles from various source systems.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Advanced Financial Closing* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Enter the URL provided by the service key under <a href="#">endpoints</a> > <i>scim2</i> without adding the path information.  For example: <code>https://afc-production-afc-api.cfapps.eu10.hana.ondemand.com</code>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user ID provided by the service key under <a href="#">uaa</a> > <i>clientid</i> .
Password	(Credential) Enter the password provided by the service key under <a href="#">uaa</a> > <i>clientsecret</i> .

Property Name	Value
OAuth2TokenServiceURL	<p>Enter the OAuth 2.0 Token Service URL provided by the service key of your SAP Advanced Financial Closing instance. It follows the pattern: <b>&lt;uaa.url&gt;/oauth/token</b>, where:</p> <ul style="list-style-type: none"> <li><b>&lt;uaa.url&gt;</b> is the URL provided by the service key under <b>uaa &gt; url</b>.</li> <li><b>/oauth/token</b> is the suffix you need to add.</li> </ul>
s4hana.afc.user.unique.attribute	<p>If Identity Provisioning tries to provision a user that already exists in the target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li><i>userName</i></li> <li><i>emails[0].value</i></li> <li><i>userName,emails[0].value</i></li> </ul>
s4hana.afc.group.unique.attribute	<p>If Identity Provisioning tries to provision a group that already exists in the target system (a conflicting group), this property defines the unique attributes by which the existing group will be searched and resolved.</p> <p>The default value is <i>displayName</i>.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map user and group attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Advanced Financial Closing* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Advanced Financial Closing system. For more information, see: [Manage Transformations \[page 1494\]](#).

[How to Manage User Access Using the SCIM API Provided](#)

[SAP Business Accelerator Hub: SAP Advanced Financial Closing](#)

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target SAP Advanced Financial Closing entity.

**Default transformation:**

## Code Syntax

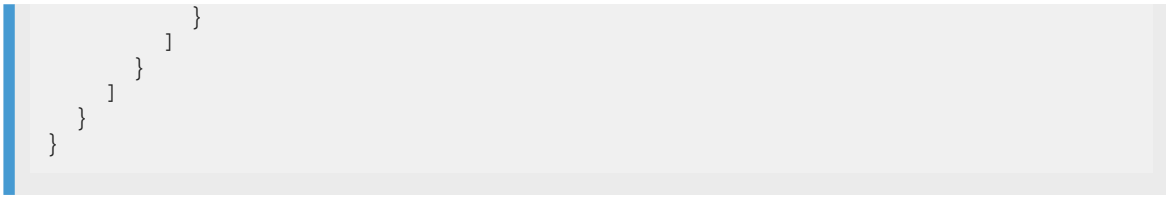
```
{
  "user": {
    "mappings": [
      {
        "constant": [
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
          "urn:ietf:params:scim:schemas:extension:sap:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUid' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUid' ]"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name.formatted",
        "targetPath": "$.name.formatted",
        "optional": true
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "condition": "$.emails[0].length() > 0",
        "targetPath": "$.emails[0].primary",
        "constant": true
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true,

```

```

        "targetPath": "$.emails",
        "functions": [
            {
                "function": "putIfAbsent",
                "key": "type",
                "defaultValue": "work"
            }
        ]
    },
    {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.roles",
        "targetPath": "$.roles",
        "preserveArrayWithSingleElement": true,
        "optional": true
    }
],
},
"group": {
    "condition": "('%s4hana.afc.group.prefix%' == 'null') ||
($.displayName =~ /%s4hana.afc.group.prefix%.*/)",
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath": "$.schemas[0]"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName",
            "functions": [
                {
                    "condition": "('%s4hana.afc.group.prefix%' != 'null') && (@
=~ /%s4hana.afc.group.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%s4hana.afc.group.prefix%",
                    "replacement": ""
                }
            ]
        },
        {
            "sourcePath": "$.members",
            "targetPath": "$.members",
            "preserveArrayWithSingleElement": true,
            "optional": true
        },
        {
            "sourcePath": "$.members[*].value",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.members[?(@.value)]",
            "functions": [
                {
                    "function": "resolveEntityIds"
                }
            ]
        }
    ]
}

```



5. Add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.4 SAP Advanced Workflow

Follow this procedure to set up a target connector for SAP Advanced Workflow.

### Prerequisites

- You have technical credentials for SAP Advanced Workflow. Note that SAP Advanced Workflow is available to SAP Commissions customers as an optional add-on. You create an Admin user in SAP Commissions, which is synchronized with SAP Advanced Workflow. For more information, see: [Adding an Admin User](#) and [Commissions User Synchronization](#).
- You have set up SSO between Identity Authentication and SAP Advanced Workflow. For more information, see [Integration with SAP IdP](#).

### Context

SAP Advanced Workflow enables you to analyse, organize, and execute business processes to connect people, data, and daily activities. Workflow provides you the tools you need to configure and customize your business processes based on your specific business needs.

After fulfilling the prerequisites, follow the procedure below to add a target system for SAP Advanced Workflow where you can provision **users** from source systems. This target system consumes Workflow SCIM API.

#### i Note

SAP Advanced Workflow does not support groups.



## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Advanced Workflow* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the API of your SAP Advanced Workflow system.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user for your SAP Advanced Workflow system.
Password	(Credential) Enter the password for your SAP Advanced Workflow user.
awf.domain	<p>The domain name is the name of your SAP Advanced Workflow tenant.</p> <p>If you don't know your tenant name, contact your supervisor or administrator, or refer to the email notification you received when your account was created.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Advanced Workflow* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Advanced Workflow. For more information, see:

[Manage Transformations \[page 1494\]](#)

The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target entity. If the entity has more than one e-mail addresses, only one of them is used. It is either the e-mail set as primary in the source system, or if no primary e-mail is set, the one that comes first is used.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY true) ||
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": ["urn:ietf:params:scim:schemas:core:2.0:User",
"urn:ietf:params:scim:schemas:extension:sap.spm.workflow:2.0:User"],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
      },
      {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
      }
    ]
  }
}
```

```

    "sourcePath": "$.timezone",
    "targetPath": "$.timezone",
    "optional": true
  }
}

```

- Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.5 SAP Analytics Cloud

Follow this procedure to set up SAP Analytics Cloud as a target system.

### Prerequisites

- In SAP Analytics Cloud, you have enabled a custom SAML Identity Provider, for which *User Attribute* is set to **Custom SAML User Mapping**. To learn how, see: [Enabling a Custom SAML Identity Provider](#)
- Add an OAuth client with authorization grant **Client Credentials**. To learn how, see: [Managing OAuth Clients and Trusted Identity Providers](#)

### Context

SAP Analytics Cloud is an all-in-one cloud product offered as software as a service for business intelligence, planning, and predictive analytics.

You can use Identity Provisioning to configure SAP Analytics Cloud as a target system where you can provision users and groups. The target system consumes SCIM 2.0 API provided by SAP Analytics Cloud. For more information, see [SAP Analytics Cloud: User and Team Provisioning API](#).

There are two versions of the SAP Analytics Cloud SCIM API. They are handled by `thesac.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on April 10, 2023), SAP Analytics Cloud SCIM API version 1 is used. This is the default value.

- When the value is set to **2** - SAP Analytics Cloud SCIM API version 2 is used. This version is released with enhancements, such as: support for patch operations.

For more information on the differences between SAP Analytics Cloud SCIM API version 1 and 2, see [Managing Users and Teams](#).

For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Analytics Cloud* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter the URL to your SAP Analytics Cloud system without adding the path information.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the client ID to retrieve the OAuth access token for SAP Analytics Cloud.
Password	(Credential) Enter the client secret to retrieve the OAuth access token for SAP Analytics Cloud.

Property Name	Description & Value
OAuth2TokenServiceURL	<p>Enter the URL of the access token provider service for your SAP Analytics Cloud instance.</p> <p>This token URL is listed in the <a href="#">OAuth Clients</a> section of the <a href="#">App Integration</a> page. For more information, refer to <a href="#">Authorize API Access for OAuth Clients</a> in <a href="#">Manage OAuth Clients</a></p>
scim.api.csrf.protection	<p>Specifies whether to fetch a CSRF token when sending requests to the system.</p> <p>This property is automatically added to the system, with default value: <b>enabled</b></p>
csrf.token.path	<p>Path which is appended to the URL to retrieve the CSRF token.</p> <p>This property is automatically added in the system, with default value: <b>/api/v1/scim/Users?count=1</b></p>
scim.user.unique.attribute	<p>This property appears by default when the system is created, and its value is set to <a href="#">userName</a>.</p> <p>It defines by which unique attribute(s) an existing user to be resolved in the event of conflicting users.</p> <p>Other possible values:</p> <ul style="list-style-type: none"> <li>• <b>emails[0].value</b></li> <li>• <b>userName,emails[0].value</b></li> <li>• <b>externalId</b>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <div> <p><b>Note</b></p> <p>If this property is missing, or you've deleted it, and the service does not find such a <a href="#">userName</a>, it will try again to resolve the conflicting user – by <a href="#">email</a>. If the second attempt for resolution is unsuccessful too, the creation of the conflicting user fails.</p> </div> <p>For more information, see: <a href="#">List of Properties [page 94]</a></p>
(Optional) sac.api.version	<p>Handles the version of SAP Analytics Cloud SCIM API.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <b>1</b> - Indicates that SAP Analytics Cloud SCIM API version 1 is used.</li> <li>• <b>2</b> - Indicates that SAP Analytics Cloud SCIM API version 2 is used.</li> </ul> <p>Default value: <b>1</b></p>

Property Name	Description & Value
(Optional) <code>sac.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable SCIM bulk operations for provisioning users. That means, the Identity Provisioning service can write, update, and delete a potentially large collection of users in a single request. For more information, see: <a href="#">SCIM Protocol: Bulk Operations</a> ➔</p> <p>If not specified, the default value is <i>false</i>.</p> <div> <p><b>i Note</b></p> <p>SCIM bulk operations are not supported for provisioning groups to SAP Analytics Cloud.</p> </div>
(Optional) <code>sac.bulk.operations.max.count</code>	<p>If you have enabled the SCIM bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p>Default value:</p> <ul style="list-style-type: none"> <li>• <b>100</b> - when using SAP Analytics Cloud SCIM API version 1.</li> <li>• <b>30</b> - when using SAP Analytics Cloud SCIM API version 2.</li> </ul> <div> <p><b>i Note</b></p> <p>The value must not exceed the number of entities defined by the SAP Analytics Cloud system as a SCIM service provider. Otherwise, the provisioning job will fail with HTTP response code 413 (<i>Payload Too Large</i>).</p> </div>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Analytics Cloud* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Analytic Cloud system. For more information, see: [Manage Transformations \[page 1494\]](#)

[SAP Analytics Cloud: User and Team Provisioning API](#)

[Managing Users and Teams → api/v1/scim](#)

[Managing Users and Teams → api/v1/scim2](#)

**Default transformation for SAP Analytics Cloud SCIM API version 1:**

## Code Syntax

```
{
  "user": {
    "condition": "($.emails[0].value EMPTY false) &&
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.schemas"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.userName"
      },
      {
        "condition": "$.emails[?(@.primary == true)].value != []",
        "sourcePath": "$.emails[?(@.primary == true)].value",
        "preserveArrayWithSingleElement": false,
        "optional": true,
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name"
      },
      {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails"
      },
      {
        "condition": "$.emails[0].length() > 0",
        "targetPath": "$.emails[0].primary",
        "constant": true
      },
      {
        "sourcePath": "$.roles",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.roles"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "optional": true,
        "targetPath": "$
['urn:scim:schemas:extension:enterprise:1.0']['manager']['managerId']",
        "functions": [
          {
```

```

        "type": "resolveEntityIds"
    }
    ]
}
],
},
"group": {
    "condition": "('%sac.group.prefix%' === 'null') || ($.displayName
    =~ /%sac.group.prefix%.*/)",
    "mappings": [
        {
            "sourcePath": "$.schemas",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.schemas"
        },
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.id",
            "scope": "createEntity",
            "functions": [
                {
                    "condition": "('%sac.group.prefix%' !== 'null') &&
                    (@ =~ /%sac.group.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%sac.group.prefix%",
                    "replacement": ""
                }
            ]
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName",
            "functions": [
                {
                    "condition": "('%sac.group.prefix%' !== 'null') &&
                    (@ =~ /%sac.group.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%sac.group.prefix%",
                    "replacement": ""
                }
            ]
        },
        {
            "sourcePath": "$.roles",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.roles"
        },
        {
            "sourcePath": "$.members[*].value",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.members[?(@.value)]",
            "functions": [
                {
                    "function": "resolveEntityIds"
                }
            ]
        }
    ]
}
]
}
}
}

```



## Default transformation for SAP Analytics Cloud SCIM API version 2:

### Code Syntax

```
{
  "user": {
    "condition": "($.emails[0].value EMPTY false) &&
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "constant": [
          "urn:sap:params:scim:schemas:extension:sac:2.0:user-
custom-parameters",
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.userName"
      },
      {
        "condition": "$.emails[?(@.primary == true)].value != []",
        "sourcePath": "$.emails[?(@.primary == true)].value",
        "preserveArrayWithSingleElement": false,
        "optional": true,
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.userName",
        "optional": true,
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "optional": true,
        "targetPath": "$.name"
      },
      {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.externalId",
        "optional": true,
        "targetPath": "$.externalId"
      },
      {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails"
      },
      {
        "condition": "$.emails[0].length() > 0",
```

```

        "constant": true,
        "targetPath": "$.emails[0].primary"
      },
      {
        "sourcePath": "$.groups[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups[?(@.value)]",
        "functions": [
          {
            "entityType": "group",
            "type": "resolveEntityIds"
          }
        ]
      }
    ],
    {
      "sourcePath": "$[ 'urn:sap:params:scim:schemas:extension:sac:2.0:user-custom-parameters' ]",
      "optional": true,
      "targetPath": "$[ 'urn:sap:params:scim:schemas:extension:sac:2.0:user-custom-parameters' ]"
    },
    {
      "sourcePath": "$.emails[0].value",
      "optional": true,

      "targetPath": "$[ 'urn:sap:params:scim:schemas:extension:sac:2.0:user-custom-parameters' ][ 'idpUserId' ]"
    },
    {
      "condition": "$.emails[?(@.primary == true)].value != []",
      "sourcePath": "$.emails[?(@.primary == true)].value",
      "preserveArrayWithSingleElement": false,
      "optional": true,

      "targetPath": "$[ 'urn:sap:params:scim:schemas:extension:sac:2.0:user-custom-parameters' ][ 'idpUserId' ]"
    },
    {
      "sourcePath": "$[ 'urn:sap:params:scim:schemas:extension:sac:2.0:user-custom-parameters' ][ 'idpUserId' ]",
      "optional": true,

      "targetPath": "$[ 'urn:sap:params:scim:schemas:extension:sac:2.0:user-custom-parameters' ][ 'idpUserId' ]"
    },
    {
      "sourcePath":
        "$[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ][ 'manager' ][ 'value' ]",
      "optional": true,
      "targetPath":
        "$[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ][ 'manager' ][ 'value' ]",
      "functions": [
        {
          "type": "resolveEntityIds"
        }
      ]
    }
  ],
  "group": {
    "condition": "( '%sac.group.prefix%' === 'null' ) || ($.displayName =~ /%sac.group.prefix%.*/)",
    "mappings": [
      {

```

```

        "constant": [
            "urn:ietf:params:scim:schemas:core:2.0:Group",
            "urn:sap:params:scim:schemas:extension:sac:2.0:group-
roles",
            "urn:sap:params:scim:schemas:extension:sac:2.0:group-
custom-parameters"
        ],
        "targetPath": "$.schemas"
    },
    {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
    },
    {
        "sourcePath": "$.displayName",
        "targetPath": "$.id",
        "scope": "createEntity",
        "functions": [
            {
                "condition": "('%sac.group.prefix%' != 'null') &&
(@ =~ /%sac.group.prefix%.*/)",
                "function": "replaceFirstString",
                "regex": "%sac.group.prefix%",
                "replacement": ""
            }
        ]
    },
    {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
            {
                "condition": "('%sac.group.prefix%' != 'null') &&
(@ =~ /%sac.group.prefix%.*/)",
                "function": "replaceFirstString",
                "regex": "%sac.group.prefix%",
                "replacement": ""
            }
        ]
    },
    {
        "sourcePath": "$.externalId",
        "optional": true,
        "targetPath": "$.externalId"
    },
    {
        "sourcePath": "$.roles",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.roles"
    },
    {
        "sourcePath": "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]",
        "functions": [
            {
                "entityType": "user",
                "type": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath": "$
['urn:sap:params:scim:schemas:extension:sac:2.0:group-roles']",
        "optional": true,

```

```

        "targetPath": "$
[ 'urn:sap:params:scim:schemas:extension:sac:2.0:group-roles' ]"
      },
      {
        "sourcePath": "$
[ 'urn:sap:params:scim:schemas:extension:sac:2.0:group-custom-parameters' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:sap:params:scim:schemas:extension:sac:2.0:group-custom-parameters' ]"
      },
      {
        "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'description' ]",
        "optional": true,
        "targetPath": "$[ 'urn:sap:params:scim:schemas:extension:sac:2.0:group-
custom-parameters' ][ 'description' ]"
      }
    ]
  }
}

```

### ⚠ Caution

When provisioning users and groups between a source system and SAP Analytics Cloud, groups are mapped to teams in SAP Analytics Cloud. Those teams can then get role assignments in SAP Analytics Cloud.

If you then run another provisioning job (**Read** or **Resync**), role assignments of SAP Analytics Cloud teams will be removed as a result of an update operation being executed. This behavior (causing permission issues for users) is expected, as SAP Analytics Cloud role assignments are not available as group parameters in some source systems, for example – Identity Authentication. To avoid this, you need to change the transformation of the [SAP Analytics Cloud](#) target system, as described in SAP Note [3027079](#).

If you try to provision groups that already exist in SAP Analytics Cloud, your provisioning job may fail with: *'The group already exists on the target system and cannot be provisioned'* error. This happens after a reset of the SAP Analytics Cloud target system or if you create a new target system connected to an existing SAP Analytics Cloud backend.

To avoid this, you have the following options:

- Delete the existing group in SAP Analytics Cloud target system.
- Adapt the SAP Analytics Cloud write transformation. Either ignore provisioning of groups or add temporary the [skipOperations](#) expression for creating groups.
- Avoid provisioning of already existing groups.


### i Note


Updating a user in SAP Analytics Cloud using SCIM API version 2 depends on whether user attributes in SAP Analytics Cloud are mapped to SAML attributes in your identity provider. If this is the case, the values of those attributes are populated by the identity provider and cannot be changed by the SAP Analytics SCIM API or the UI. For more information, see [Map SAML Attributes to Users](#)

5. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

### ⚠ Caution

The **email** attribute is unique for *SAP Analytics Cloud* but it might not be unique for other systems, such as *SAP SuccessFactors*. That's why, when you choose a source system, make sure that there are no users with duplicate e-mails. If there are such, then after the provisioning job all users with the same e-mail address will be created as a single user in *SAP Analytics Cloud*.

To learn more, see [Guided Answers: Multiple Users from a Source System Are Created as One in the Target](#) .

To learn more about excluding users from SAP Analytics Cloud target, see: [Exclude Users from Provisioning to Target Systems](#) .

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

(Guided Answers) [Multiple Users from a Source System Are Created as One in the Target](#) 

## 1.6.2.6 SAP Application Server ABAP

Follow this procedure to set up SAP Application Server ABAP (AS ABAP) as a target system.

## Prerequisites

### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)

- You have credentials of a technical user with write permissions for AS ABAP. Via this user, the Identity Provisioning service will call the ABAP public API in order to execute a number of function modules. These function modules are listed in **step 1** from the procedure below.

- You have the following role, which provides all authorizations with read and write access to user data:

#### **SAP\_BC\_JSF\_COMMUNICATION**

For more information, see: [Configuring the UME to Use an AS ABAP as Data Source](#)

## Context

SAP Application Server ABAP (AS ABAP) offers a user store and user administration capabilities for maintaining users and their authorizations for AS ABAP applications. You can configure AS ABAP as a target system so as to provision new entities from another on-premise or cloud system.

### Caution

You can't create or delete groups in AS ABAP. That means:

- On the attempt to create a group in AS ABAP, Identity Provisioning will only add new members or update existing ones. Also, when the service reads a group from a source system, there must be a group with the exact same display name (case sensitive) in the AS ABAP target system. Otherwise, an error will be thrown and the group members will not be updated.  
If group names in the source system differ from the ones in AS ABAP, you can define value mappings for these group names so the provisioning will still work. To learn how, see [Transformation Expressions \[page 330\]](#) → section **valueMapping**.
- On the attempt to delete a group in AS ABAP, Identity Provisioning will only remove its members (group assignments). And this can happen only if the relevant group assignments have been provisioned/are present in the target system.

Groups in source systems are mapped to roles in AS ABAP target systems.

### Note

Role unassignments are handled differently in AS ABAP version 7.31 and lower, and versions higher than 7.31. For example: If a child role R2 is assigned to a user – first, indirectly through a parent role R1, and then directly to the user, when the parent role is unassigned, expect the following results:

**AS ABAP 7.31 and lower:** Directly assigned parent role R1 and indirectly assigned child role R2 are unassigned. Directly assigned child role R2 is still assigned to the user.

**AS ABAP higher than 7.31:** All roles are unassigned from the user.

## Procedure

1. Open the Cloud Connector to add an access control system mapping for **AS ABAP**. This is needed to allow the Identity Provisioning service to access AS ABAP as a back-end system on the intranet. To learn how, see: [Configure Access Control \(RFC\)](#)

Go to ► [Cloud To On-Premise](#) ► [Access Control](#) ► tab and select protocol [RFC SNC](#). Then, expose the following *exact names* as accessible resources:

- PRGN\_ROLE\_GETLIST
- BAPI\_USER\_GETLIST
- BAPI\_USER\_GET\_DETAIL
- BAPI\_USER\_CREATE1
- BAPI\_USER\_ACTGROUPS\_ASSIGN
- IDENTITY\_MODIFY
- BAPI\_USER\_DELETE
- PRGN\_ACTIVITY\_GROUPS\_LOAD\_RFC

2. Open SAP BTP cockpit, and in your Identity Provisioning subaccount create a destination for the AS ABAP system. To learn how, see: [Create RFC Destinations](#)

The destination configuration is required by the Identity Provisioning service to find the back-end system to be used for writing data. It also provides the credentials of the technical user, needed for the connection to the ABAP public API.

Below are the fields you have to fill in the cockpit destination before using an AS ABAP client as a target system:

Field/Property Name	Value
<a href="#">Name</a>	Enter a destination name.
<a href="#">Type</a>	Select <a href="#">RFC</a> .
<a href="#">User</a>	Enter the user for AS ABAP.  The <a href="#">User</a> field corresponds to property <code>jco.client.user</code> in the exported RFC destination.
<a href="#">Password</a>	(Credential) Enter the password for the AS ABAP user.  The <a href="#">Password</a> field corresponds to property <code>jco.client.passwd</code> in the exported RFC destination.
<b>jco.client.client</b>	Provide the client to be used in the ABAP system. Valid format is a three-digit number.
<b>jco.destination.proxy_type</b>	Defines the proxy type of the connection you need to provide for your ABAP system.  The proxy type <a href="#">OnPremise</a> requires the Cloud Connector to access resources within your on-premise network.  Enter: <a href="#">OnPremise</a>
<b>Direct Connection</b>	
<b>jco.client.ashost</b>	Provide the virtual host entry that you have configured in the Cloud Connector → <a href="#">Access Control</a> configuration.
<b>jco.client.sysnr</b>	Provide the "system number" of the ABAP system.

Field/Property Name	Value
<b>Load Balancing Connection</b>	
<code>jco.client.mshost</code>	Represents the message server host to be used.
<code>jco.client.r3name</code>	Provide the three-character system ID of the ABAP system to be addressed.
<code>jco.client.mservt</code>	Provide the port on which the message server is listening for incoming requests. You can use this property as an alternative to <code>jco.client.r3name</code> .
<b>Optional Properties</b>	
<code>jco.destination.peak_limit</code>	The value represents the maximum number of active connections that can simultaneously be created for a destination. For example: <a href="#">10</a>
<code>jco.destination.pool_capacity</code>	The value represents the maximum number of idle connections kept open by the destination. For example: <a href="#">5</a>
<code>ips.override.existedbefore.assignments</code>	<p>This property defines whether or not the Identity Provisioning service to overwrite user/group assignments that have existed in the target system before you start provisioning entities to that system.</p> <ul style="list-style-type: none"> <li>• If you start a provisioning job without setting this property (by default, it's <i>true</i>), all assignments from the source group will overwrite the ones from the target group.</li> <li>• If you set the property to <i>false</i>, all existing assignments will be kept in the target system group, and the new ones will just be added.</li> </ul>

### Note

If connection properties, like `User` and `Password`, are configured both in the destination (SAP BTP cockpit) and on the [Properties](#) tab (Identity Provisioning User Interface), the values set in the destination are considered with higher priority.

3. Access Identity Provisioning. See: [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Application Server ABAP](#) as a target system. To learn how, see: [Add a System \[page 1477\]](#)
5. From the [Destination Name](#) dropdown, choose the RFC destination you have created in [step 2](#).
6. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Application Server ABAP](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in AS ABAP. For more information, see [Manage Transformations \[page 1494\]](#).



When AS ABAP is configured as a target system for the Identity Provisioning service, the ABAP public API is used to write the identity data retrieved from the relevant source system. During the writing process, the JSON data is following the structure of the export parameters and will turn them into list and tables. Therefore, every JSON array from the intermediate transformation will be represented as a BAPI table, and every child JSON object will be represented as a BAPI structure.

#### Default transformation:

- Use the default transformation if the version of your AS ABAP system is [7.40](#) or higher. If your system version is [7.31](#), you can still use this transformation – just apply SAP Note [1695883](#).

**Note:** If a user is *inactive* in the source system, the default transformation creates it as *inactive* (locked) in AS ABAP too.

#### Code Syntax

```
// The value of attribute entityIdTargetSystem stores the entity's
// unique ID, and then
// it's written in the target system as a username. Do not delete this
// statement!
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.USERNAME"
      },
    ],
  },

  // The userName attribute is used also as USERNAME value for the
  // internal JSON representation.
  {
    "sourcePath": "$.userName",
    "targetPath": "$.USERNAME"
  },
  {
    "sourcePath": "$.externalId",
    "optional": true,
    "targetPath": "$.ALIAS.USERALIAS",
    "defaultValue": ""
  },
  {
    "condition": "$.emails[?(@.primary == true)].value !=
[ ]",
    "sourcePath": "$.emails[?(@.primary == true)].value",
    "preserveArrayWithSingleElement": false,
    "optional": true,
    "targetPath": "$.ADDRESS.E_MAIL"
  },
  {
    "condition": "$.emails[?(@.type == 'work')].value !=
[ ]",
    "sourcePath": "$.emails[?(@.type == 'work')].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.ADDRESS.E_MAIL",
    "functions": [
      {
        "function": "elementAt",
        "index": 0
      }
    ]
  },
  {
    "sourcePath": "$.emails[*].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,
```

```

        "targetPath": "$.ADDSMTP[?(@.E_MAIL)]"
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.ADDRESS.LASTNAME"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.ADDRESS.FIRSTNAME"
      },
      {
        "sourcePath": "$.name.middleName",
        "optional": true,
        "targetPath": "$.ADDRESS.MIDDLENAME"
      },
      {
        "sourcePath": "$.nickName",
        "optional": true,
        "targetPath": "$.ADDRESS.NICKNAME"
      },
      {
        "sourcePath": "$.name.honorificPrefix",
        "optional": true,
        "targetPath": "$.ADDRESS.TITLE_P"
      },
      {
        "condition": "$.phoneNumbers[?(@.primary ==
true)].value != []",
        "sourcePath": "$.phoneNumbers[?(@.primary ==
true)].value",
        "preserveArrayWithSingleElement": false,
        "optional": true,
        "targetPath": "$.ADDRESS.TEL1_NUMBR"
      },
      {
        "condition": "$.phoneNumbers[?(@.type ==
'work')].value != []",
        "sourcePath": "$.phoneNumbers[?(@.type ==
'work')].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.ADDRESS.TEL1_NUMBR",
        "functions": [
          {
            "function": "elementAt",
            "index": 0
          }
        ]
      },
      {
        "sourcePath": "$.phoneNumbers[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.ADDTTEL[?(@.TELEPHONE)]"
      },
      {
        // The Identity Provisioning reads the locales from the source system
        // and map them as specific ABAP language codes in the target ABAP system.
        // The transformation provides an example with key = "bg", which in the
        // ABAP system is mapped as "W". The default language is en.
        // To see all languages and codes supported by AS ABAP, see the Related
        Information section below.
        "optional": true,
        "targetPath": "$.DEFAULTS.LANGU",
        "type": "valueMapping",
        "sourcePaths": [
          "$.locale"
        ]
      }
    ]
  }

```

```

    ],
    "defaultValue": "E",
    "valueMappings": [
      {
        "key": [
          "bg"
        ],
        "mappedValue": "W"
      }
    ]
  },
  {
    "sourcePath": "$.preferredLanguage",
    "optional": true,
    "targetPath": "$.ADDRESS.LANGUP_ISO",
    "functions": [
      {
        "function": "toUpperCaseString"
      }
    ]
  }
],
// The Identity Provisioning writes standard timezone codes, which are
// supported by the AS ABAP BAPI.
// However, the standard SCIM API does not support these codes, thus
// the source system can only provide values in format "<region>/<city>".
// The transformation provides an example with key = "Europe/Sofia", which
// in the target system is mapped as "EET". The default timezone is CET.
{
  "optional": true,
  "targetPath": "$.LOGONDATA.TZONE",
  "type": "valueMapping",
  "sourcePaths": [
    "$.timezone"
  ],
  "defaultValue": "CET",
  "valueMappings": [
    {
      "key": [
        "Europe/Sofia"
      ],
      "mappedValue": "EET"
    }
  ]
},
{
  "targetPath": "$.PASSWORD.BAPIPWD",
  "scope": "createEntity",
  "functions": [
    {
      "function": "randomPassword",
      "passwordLength": 24,
      "minimumNumberOfLowercaseLetters": 1,
      "minimumNumberOfUppercaseLetters": 1,
      "minimumNumberOfDigits": 1,
      "minimumNumberOfSpecialSymbols": 0
    }
  ]
},
{
  "constant": "updateEntity",
  "targetVariable": "operationTypeVariable"
},
{
  "constant": "createEntity",
  "targetVariable": "operationTypeVariable",
  "scope": "createEntity"
},
{

```

```

        "condition": "$.active == false && '{operationTypeVariable}' == 'createEntity'",
        "constant": "X",
        "targetPath": "$.LOCK_LOCALLY"
      },
      {
        "condition": "'${operationTypeVariable}' == 'updateEntity'",
        "constant": "U",
        "targetPath": "$.LOCK"
      },
      {
        "condition": "$.active == false && '{operationTypeVariable}' == 'updateEntity'",
        "constant": "L",
        "targetPath": "$.LOCK"
      }
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": $.displayName,
        "targetVariable": entityIdTargetSystem,
        "scope": "createEntity"
      },
      {
        "sourcePath": $.displayName,
        "targetPath": $.ROLE_NAME
      },
      {
        "sourcePath": $.members[*].value,
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": $.USERLIST[?(@.USERNAME)]
      }
    ]
  }
}

```

- If your AS ABAP version is [7.30](#) or lower, use the transformation below.

**Note:** This transformation creates all users as *active* (unlocked) in AS ABAP, regardless if they are *active* or *inactive* in the source system.

#### Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": $.USERNAME
      },
      {
        "sourcePath": $.userName,
        "targetPath": $.USERNAME
      },
      {
        "sourcePath": $.externalId,
        "optional": true,
        "targetPath": $.ALIAS.USERALIAS,
        "defaultValue": ""
      },
      {
        "condition": "$.emails[?(@.primary == true)].value !=
    ]",

```

```

        "sourcePath": "$.emails[?(@.primary == true)].value",
        "preserveArrayWithSingleElement": false,
        "optional": true,
        "targetPath": "$.ADDRESS.E_MAIL"
    },
    {
        "condition": "$.emails[?(@.type == 'work')].value !=
[]",
        "sourcePath": "$.emails[?(@.type == 'work')].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.ADDRESS.E_MAIL",
        "functions": [
            {
                "function": "elementAt",
                "index": 0
            }
        ]
    },
    {
        "sourcePath": "$.emails[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.ADDSMTP[?(@.E_MAIL)]"
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.ADDRESS.LASTNAME"
    },
    {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.ADDRESS.FIRSTNAME"
    },
    {
        "sourcePath": "$.name.middleName",
        "optional": true,
        "targetPath": "$.ADDRESS.MIDDLENAME"
    },
    {
        "sourcePath": "$.nickName",
        "optional": true,
        "targetPath": "$.ADDRESS.NICKNAME"
    },
    {
        "sourcePath": "$.name.honorificPrefix",
        "optional": true,
        "targetPath": "$.ADDRESS.TITLE_P"
    },
    {
        "condition": "$.phoneNumbers[?(@.primary ==
true)].value != []",
        "sourcePath": "$.phoneNumbers[?(@.primary ==
true)].value",
        "preserveArrayWithSingleElement": false,
        "optional": true,
        "targetPath": "$.ADDRESS.TEL1_NUMBR"
    },
    {
        "condition": "$.phoneNumbers[?(@.type ==
'work')].value != []",
        "sourcePath": "$.phoneNumbers[?(@.type ==
'work')].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.ADDRESS.TEL1_NUMBR",
        "functions": [
            {

```

```


        "function": "elementAt",
        "index": 0
    }
]
},
{
    "sourcePath": "$.phoneNumbers[*].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.ADDTTEL[?(@.TELEPHONE)]"
},
{
    "optional": true,
    "targetPath": "$.DEFAULTS.LANGU",
    "type": "valueMapping",
    "sourcePaths": [
        "$.locale"
    ],
    "defaultValue": "E",
    "valueMappings": [
        {
            "key": [
                "bg"
            ],
            "mappedValue": "W"
        }
    ]
},
{
    "sourcePath": "$.preferredLanguage",
    "optional": true,
    "targetPath": "$.ADDRESS.LANGUP_ISO",
    "functions": [
        {
            "function": "toUpperCaseString"
        }
    ]
},
{
    "optional": true,
    "targetPath": "$.LOGONDATA.TZONE",
    "type": "valueMapping",
    "sourcePaths": [
        "$.timezone"
    ],
    "defaultValue": "CET",
    "valueMappings": [
        {
            "key": [
                "Europe/Sofia"
            ],
            "mappedValue": "EET"
        }
    ]
},
{
    "targetPath": "$.PASSWORD.BAPIPWD",
    "scope": "createEntity",
    "functions": [
        {
            "function": "randomPassword",
            "passwordLength": 24,
            "minimumNumberOfLowercaseLetters": 1,
            "minimumNumberOfUppercaseLetters": 1,
            "minimumNumberOfDigits": 1,
            "minimumNumberOfSpecialSymbols": 0
        }
    ]
}
]

```

```

    },
    {
      "constant": "updateEntity",
      "targetVariable": "operationTypeVariable"
    },
    {
      "constant": "createEntity",
      "targetVariable": "operationTypeVariable",
      "scope": "createEntity"
    },
    {
      "condition": "'${operationTypeVariable}' ==
'updateEntity'",
      "constant": "U",
      "targetPath": "$.LOCK"
    },
    {
      "condition": "$.active == false && '${
operationTypeVariable}' == 'updateEntity'",
      "constant": "L",
      "targetPath": "$.LOCK"
    }
  ]
},
"group": {
  "mappings": [
    {
      "sourcePath": "$.displayName",
      "targetVariable": "entityIdTargetSystem",
      "scope": "createEntity"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.ROLE_NAME"
    },
    {
      "sourcePath": "$.members[*].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.USERLIST[?(@.USERNAME)]"
    }
  ]
}
}

```

- Default transformation supporting User UUID attribute:  
The following SAP Note must be implemented in the SAP AS ABAP [3003462](#)  *Interface enhancement for global user ID.*

#### Code Syntax

```

{
  "user": {
    "condition": "($.emails EMPTY true) ||
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.USERNAME"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.USERNAME"
      },
      {
        "sourcePath": "$.externalId",

```

```

        "optional": true,
        "targetPath": "$.ALIAS.USERALIAS",
        "defaultValue": ""
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true,
        "targetPath": "$.SAPUSER_UUID.SAP_UID"
    },
    {
        "condition": "$.emails[?(@.primary == true)].value != []",
        "sourcePath": "$.emails[?(@.primary == true)].value",
        "preserveArrayWithSingleElement": false,
        "optional": true,
        "targetPath": "$.ADDRESS.E_MAIL"
    },
    {
        "condition": "$.emails[?(@.type == 'work')].value != []",
        "sourcePath": "$.emails[?(@.type == 'work')].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.ADDRESS.E_MAIL",
        "functions": [
            {
                "function": "elementAt",
                "index": 0
            }
        ]
    },
    {
        "sourcePath": "$.emails[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.ADDSMTP[?(@.E_MAIL)]"
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.ADDRESS.LASTNAME"
    },
    {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.ADDRESS.FIRSTNAME"
    },
    {
        "sourcePath": "$.name.middleName",
        "optional": true,
        "targetPath": "$.ADDRESS.MIDDLENAME"
    },
    {
        "sourcePath": "$.nickName",
        "optional": true,
        "targetPath": "$.ADDRESS.NICKNAME"
    },
    {
        "sourcePath": "$.name.honorificPrefix",
        "optional": true,
        "targetPath": "$.ADDRESS.TITLE_P"
    },
    {
        "condition": "$.phoneNumbers[?(@.primary == true)].value != []",
        "sourcePath": "$.phoneNumbers[?(@.primary == true)].value",
        "preserveArrayWithSingleElement": false,
        "optional": true,
        "targetPath": "$.ADDRESS.TEL1_NUMBR"
    },
    {

```



```

"condition": "$.phoneNumbers[?(@.type == 'work')].value != []",
"sourcePath": "$.phoneNumbers[?(@.type == 'work')].value",
"preserveArrayWithSingleElement": true,
"optional": true,
"targetPath": "$.ADDRESS.TEL1_NUMBR",
"functions": [
  {
    "function": "elementAt",
    "index": 0
  }
]
},
{
  "sourcePath": "$.phoneNumbers[*].value",
  "preserveArrayWithSingleElement": true,
  "optional": true,
  "targetPath": "$.ADDTEL[?(@.TELEPHONE)]"
},
{
  "type": "valueMapping",
  "sourcePaths": [
    "$.locale"
  ],
  "optional": true,
  "targetPath": "$.DEFAULTS.LANGU",
  "defaultValue": "E",
  "valueMappings": [
    {
      "key": [
        "bg"
      ],
      "mappedValue": "W"
    }
  ]
},
{
  "sourcePath": "$.preferredLanguage",
  "optional": true,
  "targetPath": "$.ADDRESS.LANGUP_ISO",
  "functions": [
    {
      "function": "toUpperCaseString"
    }
  ]
},
{
  "type": "valueMapping",
  "sourcePaths": [
    "$.timezone"
  ],
  "optional": true,
  "targetPath": "$.LOGONDATA.TZONE",
  "defaultValue": "CET",
  "valueMappings": [
    {
      "key": [
        "Europe/Sofia"
      ],
      "mappedValue": "EET"
    }
  ]
},
{
  "targetPath": "$.PASSWORD.BAPIPWD",
  "scope": "createEntity",
  "functions": [
    {
      "function": "randomPassword",

```

```

        "passwordLength": 24,
        "minimumNumberOfLowercaseLetters": 1,
        "minimumNumberOfUppercaseLetters": 1,
        "minimumNumberOfDigits": 1,
        "minimumNumberOfSpecialSymbols": 0
    }
}
},
{
    "constant": "updateEntity",
    "targetVariable": "operationTypeVariable"
},
{
    "constant": "createEntity",
    "targetVariable": "operationTypeVariable",
    "scope": "createEntity"
},
{
    "condition": "$.active == false && '${operationTypeVariable}'
== 'createEntity'",
    "constant": "X",
    "targetPath": "$.LOCK_LOCALLY"
},
{
    "condition": "'${operationTypeVariable}' == 'updateEntity'",
    "constant": "U",
    "targetPath": "$.LOCK"
},
{
    "condition": "$.active == false && '${operationTypeVariable}'
== 'updateEntity'",
    "constant": "L",
    "targetPath": "$.LOCK"
}
]
},
"group": {
    "condition": "('%abap.role.prefix%' === 'null') || ($.displayName
== ~/abap.role.prefix%.*)/",
    "mappings": [
        {
            "scope": "createEntity",
            "sourcePath": "$.displayName",
            "targetVariable": "entityIdTargetSystem",
            "functions": [
                {
                    "condition": "('%abap.role.prefix%' !== 'null') && (@ == ~/
%abap.role.prefix%.*)/",
                    "function": "replaceFirstString",
                    "regex": "%abap.role.prefix%",
                    "replacement": ""
                }
            ]
        }
    ],
    {
        "sourcePath": "$.displayName",
        "targetPath": "$.ROLE_NAME",
        "functions": [
            {
                "condition": "('%abap.role.prefix%' !== 'null') && (@ == ~/
%abap.role.prefix%.*)/",
                "function": "replaceFirstString",
                "regex": "%abap.role.prefix%",
                "replacement": ""
            }
        ]
    }
}
},

```

```

    "sourcePath": "$.members[*].value",
    "preserveArrayWithSingleElement": true,
    "targetPath": "$.USERLIST[?(@.USERNAME)]",
    "optional": true,
    "functions": [
      {
        "function": "resolveEntityIds"
      }
    ]
  }
}

```

- Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[ABAP: Supported Languages and Code Pages](#)

## 1.6.2.7 SAP Ariba Applications

Follow this procedure to set up SAP Ariba Applications as a target system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You have created a client application on [SAP Ariba APIs Portal](#)  that needs to be enabled for Identity Provisioning.

## i Note

If you don't have an account on SAP Ariba Developer Portal, then ask your **Designated Support Contact** (DSC) to submit a [request for an account](#). To find your DSC person, see: [How can I see my company's Basic users and Designated Support Contacts \(DSC\)](#) 📄

2. Provide your DSC person with your SAP Ariba **realm name**, **application name**, and **application key**. You have already created the application name along with the application key on [step 1](#). To find your realm name, login to your SAP Ariba system – it's part of your login URL, as shown in the following examples.
  - *SAP Ariba Buyer* example: <https://s1.ariba.com/Buyer/Main/ad/loginPage/...&realm=mycompany-t>
  - *SAP Ariba Sourcing* example: <http://mycompany.sourcing.ariba.com/>
3. Ask your DSC person to submit a service request for you to [SAP Ariba Support](#) for component **BNS-ARI-SS-API**, requesting the client application to be enabled for Identity Provisioning. Request your DSC person to mention the following details in the service request:
  - Application name
  - Application key
  - Realm name
4. When your application is enabled, you can login to [SAP Ariba APIs Portal](#) 📄, find your application, and generate a new OAuth secret for it. To learn how, see: [How to generate the OAuth Secret and Base64 Encoded Client and secret](#)
5. To configure your [SAP Ariba Applications](#) provisioning system (see the procedure below), you will need to map your SAP Ariba application parameters to the relevant Identity Provisioning properties. The property mapping between the two systems is as follows:

SAP Ariba	Identity Provisioning	Values
SCIM API URL	URL	Examples: <ul style="list-style-type: none"><li>• US: <a href="https://openapi.ariba.com">https://openapi.ariba.com</a></li><li>• Europe: <a href="https://eu.openapi.ariba.com">https://eu.openapi.ariba.com</a></li><li>• UAE: <a href="https://mn1.openapi.ariba.com">https://mn1.openapi.ariba.com</a></li></ul>
SAP Ariba OAuth 2.0 Token URL	OAuth2TokenServiceURL	Examples: <ul style="list-style-type: none"><li>• US: <a href="https://api.ariba.com/v2/oauth/token">https://api.ariba.com/v2/oauth/token</a></li><li>• Europe: <a href="https://api-eu.ariba.com/v2/oauth/token">https://api-eu.ariba.com/v2/oauth/token</a></li><li>• UAE: <a href="https://api.mn1.ariba.com/v2/oauth/token">https://api.mn1.ariba.com/v2/oauth/token</a></li></ul>
OAuth Client ID	User	Alphanumeric string Example: <b>aaaa12345-1111-3333-cccc-1234567890</b>
OAuth Secret	Password	Alphanumeric string

SAP Ariba	Identity Provisioning	Values
		Example: <b>aaaGGG1eee12abcdefGHIJK123lmnopTTT</b>
Application key	<code>ariba.applications.api.key</code>	Alphanumeric string Example: <b>123abc123XYZ000abc123ABC012345</b>
AN-ID	<code>ariba.applications.realm.id</code>	AN<numeric_string> Example: <b>AN000111222333</b>

## Context

After fulfilling the prerequisites, you can create an SAP Ariba Applications target system to provision users and groups.

These target systems consume SCIM 2.0 API provided by SAP Ariba Applications. For more information about the SAP Ariba SCIM API scope of support, see [3228340](#) .

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Ariba Applications* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter the SCIM API URL for your SAP Ariba application (see the <b>Prerequisites</b> section).
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client ID (see the <b>Prerequisites</b> section).
Password	(Credential) Enter the OAuth Secret (see the <b>Prerequisites</b> section).
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL (see the <b>Prerequisites</b> section).
ariba.applications.api.key	(Credential) Enter your application key (see the <b>Prerequisites</b> section).
ariba.applications.realm.id	Enter your AN-ID (see the <b>Prerequisites</b> section).

Exemplary destination (property configuration):

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://openapi.ariba.com
User=aaaa12345-1111-3333-cccc-1234567890
Password=*****
OAuth2TokenServiceURL=https://api.ariba.com/v2/oauth/token
ariba.applications.api.key=123abc123XYZ000abc123ABC012345
ariba.applications.realm.id=AN000111222333
```

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Ariba Applications](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Ariba Applications. For more information, see:

[Manage Transformations \[page 1494\]](#)

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

#### Default transformation:

##### Code Syntax

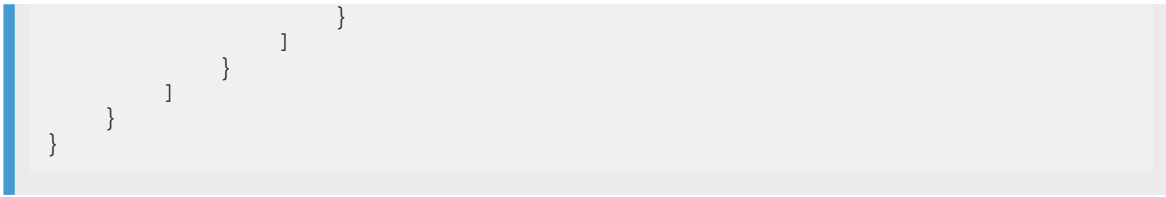
```
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": "$.schemas[1]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:sap:2.0:User",
        "targetPath": "$.schemas[2]"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "condition": "$.emails[0].length() > 0",
        "constant": true,
        "targetPath": "$.emails[0].primary"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
      },
      {
        "sourcePath": "$.locale",
        "optional": true,
        "targetPath": "$.locale"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      }
    ]
  }
}
```

```

        {
            "sourcePath": "$.timezone",
            "optional": true,
            "targetPath": "$.timezone"
        },
        {
            "sourcePath": "$.phoneNumbers",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.phoneNumbers",
            "functions": [
                {
                    "function": "putIfAbsent",
                    "key": "type",
                    "defaultValue": "work"
                }
            ]
        }
    ],
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "optional": true,
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    }
]
},
"group": {
    "condition": "('%ariba.applications.group.prefix%' === 'null') ||
($.displayName =~ /%ariba.applications.group.prefix%.*/)",
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath": "$.schemas[0]"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName",
            "functions": [
                {
                    "condition":
"$('%ariba.applications.group.prefix%' !== 'null') && (@ =~ /
%ariba.applications.group.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%ariba.applications.group.prefix%",
                    "replacement": ""
                }
            ]
        }
    ]
},
{
    "sourcePath": "$.members[*].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members[?(@.value)]",
    "functions": [
        {
            "type": "resolveEntityIds"
        }
    ]
}

```





5. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe to the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during your jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[The SAP Ariba developer portal](#)

[Video: Create application and API approval process](#) 📺

## 1.6.2.8 SAP BTP ABAP environment

Follow this procedure to set up SAP BTP ABAP environment as a target system.

## Prerequisites

To establish the connection between Identity Provisioning and SAP BTP ABAP environment, you need to set up the communication (user, system and arrangement) on SAP BTP ABAP environment. You can do it now (as a prerequisite) or in the process of configuring SAP BTP ABAP environment as a target system, as described in step 3.

## Context

You can use SAP BTP ABAP environment as a target system to provision entities from a certain source system. This scenario supports writing **users** and **assignments**. In SAP BTP ABAP environment, groups correspond to roles, thus group members are user assignments of a role.

## i Note

Identity Provisioning cannot create and delete roles in SAP BTP ABAP environment target system. It can only create, update and delete user assignments of a role. Therefore, roles must have been created in SAP BTP ABAP environment target system before you run a provisioning job.

For example, if you try to create or delete a role in SAP BTP ABAP environment, Identity Provisioning will only add or remove the user assignments of that role, respectively.




The Identity Provisioning service manages the complete set of **business partners** and their relevant **business users** (Employee).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP BTP ABAP environment* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP BTP ABAP environment and configure your authentication method (basic or certificate-based).

## i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP BTP ABAP environment target system, select the *Certificate* tab and choose  [Generate](#)  [Download](#) , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP BTP ABAP environment backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose *Scenario ID* SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

For more information, see [Maintain a Communication Arrangement for Inbound Communication](#) .

### i Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.


If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the API URL to your SAP BTP ABAP environment system.</p> <p>You can take the URL from the communication scenario SAP_COM_0193.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>

Property Name	Description & Value
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>
a4c.user.unique.attribute	<p>If Identity Provisioning tries to provision a user that already exists in the SAP BTP ABAP environment target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property does not appear in the UI during system creation. Its default value is <a href="#">personExternalID</a>. That means, if the service finds an existing user by a <a href="#">personExternalID</a>, it updates this user with the data of the conflicting one. If a user with such a <a href="#">personExternalID</a> is not found, the creation of the conflicting user fails.</li> <li>• Value = <a href="#">emails[0].value</a>. If the service finds an existing user matching both unique attributes <a href="#">email</a> and <a href="#">personExternalID</a>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <a href="#">email</a>, the update of the existing user fails. If a user with such <a href="#">email</a> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <a href="#">personExternalID</a></li> <li>• <a href="#">emails[0].value</a></li> </ul> <p>Default value: <a href="#">personExternalID</a></p>

Property Name	Description & Value
<code>a4c.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP BTP ABAP environment target system.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the target system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the target system.</li> </ul> <p>Default value (if the property is missing during system creation): <i>true</i></p> <p>Default value (if the property appears during system creation): <i>false</i></p> <p>See also: <a href="#">Extended Explanation of the *.user.roles.override Properties</a> </p>
(Optional) <code>a4c.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request. For more information, see: <a href="#">APIs for Business User Management</a></p> <p>If not specified, the default value is <i>false</i>.</p>
(Optional) <code>a4c.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p>The default value, if not specified, is <i>20</i>.</p> <p>The maximum value is <b>100</b>. If you enter a number larger than 100, the service will replace it with the default value (20).</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP

Authentication=BasicAuthentication

ProxyType=Internet

URL=https://12345-aaaaa-3333.abap.hana.ondemand.com

User=MyABAPEnvUser

Password=*****

ips.date.variable.format=yyyy-MM-dd

a4c.user.roles.override=false

a4c.support.bulk.operation = true

a4c.bulk.operations.max.count = 50
```

---

#### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP BTP ABAP environment](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in SAP BTP ABAP environment. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY false) &&
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.personExternalID"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.personExternalID",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath": "$.personExternalID",
        "optional": true
      }
    ]
  }
}
```

```

        "targetPath": "$.personID",
        "sourceVariable": "entityIdTargetSystem"
    },
    {
        "targetPath": "$.markedForArchivingIndicator",
        "constant": "false"
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true,
        "targetPath": "$.user.globalUserID"
    },
    {
        "type": "valueMapping",
        "sourcePaths": [
            "$.userType"
        ],
        "targetPath": "$.businessPartnerRoleCode",
        "defaultValue": "BUP003",
        "valueMappings": [
            {
                "key": [
                    "Employee"
                ],
                "mappedValue": "BUP003"
            }
        ]
    },
    {
        "scope": "createEntity",
        "targetPath": "$.validityPeriod.startDate",
        "sourceVariable": "currentDate"
    },
    {
        "scope": "createEntity",
        "targetPath": "$.validityPeriod.endDate",
        "constant": "9999-12-31"
    },
    {
        "scope": "createEntity",
        "sourceVariable": "currentDate",
        "targetPath": "$.user.validityPeriod.startDate"
    },
    {
        "scope": "createEntity",
        "constant": "9999-12-31",
        "targetPath": "$.user.validityPeriod.endDate"
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.personalInformation.firstName",
        "optional": true
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.personalInformation.lastName"
    },
    {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.personalInformation.middleName",
        "optional": true
    },
    {
        "sourcePath": "$.name.formatted",
        "targetPath": "$.personalInformation.personFullName",
        "optional": true
    },
    },

```

```

        {
            "sourcePath": "$.userName",
            "targetPath": "$.user.userName"
        },
        {
            "sourcePath": "$.locale",
            "targetPath": "$.user.logonLanguageCode",
            "optional": true
        },
        {
            "sourcePath": "$.emails[0].value",
            "targetPath": "$.workplaceInformation.emailAddress"
        },
        {
            "condition": "$.active == false",
            "targetPath": "$.user.lockedIndicator",
            "constant": "true"
        }
    ]
},
"group": {
    "condition": "('%a4c.roles.prefix%' === 'null') || ($.displayName =~ /%a4c.roles.prefix%.*/)",
    "mappings": [
        {
            "sourcePath": "$.displayName",
            "functions": [
                {
                    "condition": "('%a4c.roles.prefix%' !== 'null') && (@ =~ /%a4c.roles.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%a4c.roles.prefix%",
                    "replacement": ""
                }
            ],
            "targetVariable": "entityIdTargetSystem",
            "scope": "createEntity"
        },
        {
            "sourcePath": "$.displayName",
            "functions": [
                {
                    "condition": "('%a4c.roles.prefix%' !== 'null') && (@ =~ /%a4c.roles.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%a4c.roles.prefix%",
                    "replacement": ""
                }
            ],
            "targetPath": "$.displayName"
        },
        {
            "sourcePath": "$.members[*].value",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.members[?(@.value)]",
            "functions": [
                {
                    "function": "resolveEntityIds"
                }
            ]
        }
    ]
}
]
}
}

```



See also: [Extended Explanation of the \\*user.roles.override Properties](#) 

6. Now, add a source system from which to read users and roles. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA Cloud Documentation](#)

### 1.6.2.9 SAP BTP Account Members (Neo)

Follow this procedure to set up your SAP Business Technology Platform as a target system, to which you can provision and write members in your Neo account.

## Prerequisites

You have created a new Platform API OAuth client, with API [Account Member Management](#) and scopes [Manage Account Members](#) and [Read Account Members](#).

Save the [Client ID](#) and [Client Secret](#) as you'll need them when you configure your target system. Make sure you save the client secret as you cannot retrieve it later.

For more information, see [Create a Platform API Client](#).

## Context

The Identity Provisioning service helps companies to automatically manage the user-to-platform roles assignments for SAP Business Technology Platform subaccounts. For this aim, the service reuses data from an active company user store. For this scenario, as a target system are used SAP Business Technology Platform subaccounts. A source system can be a solution supported by the Identity Provisioning service with read access for user artifacts.

This provisioning scenario is based on the Platform Authorization Management API of SAP BTP. For more information, see [Platform Authorization Management API](#).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP BTP Account Members (Neo)* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter: <b>https://api.&lt;SAP_BTP_host&gt;/authorization/v1/platform/accounts/&lt;SAP_BTP_account&gt;</b>  Examples: <ul style="list-style-type: none"><li>• (Europe – Rot) <a href="https://api.hana.ondemand.com/authorization/v1/platform/accounts/abc123xyz">https://api.hana.ondemand.com/authorization/v1/platform/accounts/abc123xyz</a></li><li>• (Japan – Tokyo) <a href="https://api.jp1.hana.ondemand.com/authorization/v1/platform/accounts/abc123xyz">https://api.jp1.hana.ondemand.com/authorization/v1/platform/accounts/abc123xyz</a></li></ul>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the Client ID of the new Platform API OAuth client created for the <b>Account Member Management</b> API (see the prerequisites).
Password	(Credential) Enter the Client Secret of the new Platform API OAuth client created for the <b>Account Member Management</b> API (see the prerequisites).

Property Name	Description & Value
OAuth2TokenServiceURL	<p>Enter: <b>https://api.&lt;SAP_CP_host&gt;/oauth2/apitoken/v1</b></p> <p>Examples:</p> <ul style="list-style-type: none"> <li>(Europe – Rot) <a href="https://api.hana.ondemand.com/oauth2/apitoken/v1">https://api.hana.ondemand.com/oauth2/apitoken/v1</a></li> <li>(US East – Sterling) <a href="https://api.us3.hana.ondemand.com/oauth2/apitoken/v1">https://api.us3.hana.ondemand.com/oauth2/apitoken/v1</a></li> </ul>
scp.user.userbase	<p>This property specifies the host to the identity provider to be used with this proxy system. All provisioned users can be authenticated only by this identity provider. Default value: <a href="https://account.sap.com">account.sap.com</a></p> <p>If you use another IdP, enter its value as configured in the SAP BTP cockpit. For example:</p> <p><b>&lt;account_ID&gt;.accounts.ondemand.com</b></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP BTP Account Members](#) target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

Using the default transformation, all groups that are available in the source system and their respective members (as identifiers) are assigned as platform roles to the users. These users are added to the SAP Business Technology Platform subaccount via the target system.

You can change the default transformation mapping rules to reflect your current setup of entities in your target system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP BTP: Authorization Management API](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      }
    ]
  }
}
```

```

// The roles in the conditions below are placeholder names mapped to the
// SAP BTP roles. You can replace these names with the actual role names used
// in your source system.
// You can delete some of these roles (if redundant), or add new mappings
// for additional roles.
{
  "condition": " (('scp.group.prefix%' === 'null')
&& ($.groups[*].value contains 'AccountAdministrator')) ||
(('scp.group.prefix%' !== 'null') && ($.groups[*].value contains
'scp.group.prefix%AccountAdministrator'))",
  "constant": "AccountAdministrator",
  "targetPath": "$.roles[0].value"
},
{
  "condition": " (('scp.group.prefix%' === 'null') &&
($.groups[*].value contains 'Developer')) || (('scp.group.prefix%' !==
'null') && ($.groups[*].value contains 'scp.group.prefix%Developer'))",
  "constant": "Developer",
  "targetPath": "$.roles[1].value"
},
{
  "condition": " (('scp.group.prefix%' === 'null') &&
($.groups[*].value contains 'CloudConnectorAdministrator')) ||
(('scp.group.prefix%' !== 'null') && ($.groups[*].value contains
'scp.group.prefix%CloudConnectorAdministrator'))",
  "constant": "CloudConnectorAdministrator",
  "targetPath": "$.roles[2].value"
},
{
  "condition": " (('scp.group.prefix%' === 'null') &&
($.groups[*].value contains 'ApplicationUserAdministrator')) ||
(('scp.group.prefix%' !== 'null') && ($.groups[*].value contains
'scp.group.prefix%ApplicationUserAdministrator'))",
  "constant": "ApplicationUserAdministrator",
  "targetPath": "$.roles[3].value"
},
{
  "condition": " (('scp.group.prefix%' === 'null') &&
($.groups[*].value contains 'ReadOnly')) || (('scp.group.prefix%' !==
'null') && ($.groups[*].value contains 'scp.group.prefix%ReadOnly'))",
  "constant": "ReadOnly",
  "targetPath": "$.roles[4].value"
},
{
  "constant": "%scp.user.userbase%",
  "optional": true,
  "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:UserExt']['userbase']"
},
{
  "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
  "targetPath": "$.schemas[0]"
},
{
  "constant":
"urn:sap:cloud:scim:schemas:extension:custom:2.0:UserExt",
  "targetPath": "$.schemas[1]"
}
],
"group": {
  "condition": " (('scp.group.prefix%' === 'null') || ($.displayName =~ /
%scp.group.prefix%.*/))",
  "skipOperations": [
    "delete"
  ],
  "mappings": [
    {

```

```

        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "('%scp.group.prefix%' !== 'null') && (@ =~ /%scp.group.prefix%.*/)",
            "function": "replaceFirstString",
            "regex": "%scp.group.prefix%",
            "replacement": ""
          }
        ]
      }
    ],
    {
      "optional": true,
      "preserveArrayWithSingleElement": true,
      "sourcePath": "$.members[*].value",
      "targetPath": "$.members[?(@.value)]",
      "functions": [
        {
          "type": "resolveEntityIds"
        }
      ]
    }
  ]
}

```

- Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[What is SAP BTP](#)

## 1.6.2.10 SAP BTP Java/HTML5 apps (Neo)

Follow this procedure to set up SAP Business Technology Platform as a target system, to which you can provision user-to-groups assignments for Java/HTML5 applications that run on SAP BTP, Neo environment.

### Prerequisites

You have created a new Platform API OAuth client, with [Authorization Management](#) scopes. Save the [Client ID](#) and [Client Secret](#) as you'll need them when you configure your target system. Make sure you save the client secret as you cannot retrieve it later.

For more information, see [Create a Platform API Client](#).

### Context

The Identity Provisioning service helps companies to automatically manage the user-to-groups assignments for Java/HTML5 applications running on SAP BTP, Neo environment. For this aim, the service reuses data from an active company user store. For this scenario, SAP BTP is the target system. The source system can be a solution supported by the Identity Provisioning service with read access for user and group artifacts.

This provisioning scenario is based on the Authorization Management REST API of SAP BTP. For more information, see [Using the Authorization Management REST API](#).

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add [SAP BTP Java/HTML5 apps \(Neo\)](#) as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the [Properties](#) tab to configure the connection settings for your system.

#### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter: <a href="#">https://api.&lt;SAP_BTP_host&gt;/authorization/v1/accounts/&lt;SAP_BTP_account&gt;</a>  Examples: <ul style="list-style-type: none"> <li>• (Europe – Rot) <a href="#">https://api.hana.ondemand.com/authorization/v1/accounts/abc123xyz</a></li> <li>• (Japan – Tokyo) <a href="#">https://api.jp1.hana.ondemand.com/authorization/v1/accounts/abc123xyz</a></li> </ul>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the Client ID of the new Platform API OAuth client created for the Authorization Management API (see the prerequisites).
Password	(Credential) Enter the Client Secret of the new Platform API OAuth client created for the Authorization Management API (see the prerequisites).
OAuth2TokenServiceURL	Enter: <a href="#">https://api.&lt;SAP_BTP_host&gt;/oauth2/apitoken/v1</a>  Examples: <ul style="list-style-type: none"> <li>• (Europe – Rot) <a href="#">https://api.hana.ondemand.com/oauth2/apitoken/v1</a></li> <li>• (US East – Sterling) <a href="#">https://api.us3.hana.ondemand.com/oauth2/apitoken/v1</a></li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP BTP Java/HTML5 apps \(Neo\)](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

Using the default transformation, all groups that are available in the source system and their respective members (as identifiers) will be created as groups and members in the SAP BTP account. In the target system, they will be configured and assigned to the same list of users (as identities) that are available as members for these roles in the source system.

### i Note

Make sure the [group](#) mapping in the source system is enabled. If it contains **"ignore": true**, change it to **"ignore": false**.

You can change the default transformation mapping rules to reflect your current setup of entities in your target system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP BTP: Authorization Management API](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "ignore": true,
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdTargetSystem"
      }
    ]
  },
  "group": {
    "condition": "('%hcp.group.prefix%' === 'null') || ($.displayName =~ /%hcp.group.prefix%.*/)",
    "mappings": [
      // Attribute entityIdTargetSystem stores the displayName attribute as a
      // unique value of the group.
      {
        "sourcePath": "$.displayName",
        "targetVariable": "entityIdTargetSystem",
        "functions": [
          {
            "condition": "('%hcp.group.prefix%' !== 'null') && (@
            =~ /%hcp.group.prefix%.*/)",
            "function": "replaceFirstString",
            "regex": "%hcp.group.prefix%",
            "replacement": ""
          }
        ]
      }
    ]
  },
  // All members of a source group will be transformed, by default, into
  // users for a new group.
  // This group will be created in SAP BTP when the JSON data is prepared to
  // be sent to the target system.
  {
    "sourcePath": "$.members[*].value",
    "optional": true,
    "targetPath": "$.users"
  }
]
```

5. Now, add a source system from which to read groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).



2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[What is SAP BTP](#)

### 1.6.2.11 SAP BTP XS Advanced UAA (Cloud Foundry)

Follow this procedure to set up the SAP BTP XS Advanced UAA (running on SAP BTP, Cloud Foundry environment) as a target system.

## Prerequisites

- You have a technical database user for a SAP BTP XS Advanced UAA system with write access permissions. To learn how, see: [Get API Access](#)
- You have a subaccount on SAP BTP, Cloud Foundry environment, and Cloud Foundry applications for which you have been subscribed.
- Since OAuth is used for authentication of your service instance, you need to generate a service key for the service instance, and then retrieve this service key with OAuth 2.0 client credentials (client ID and secret). You'll use them when creating a destination (or specifying the Identity Provisioning connection properties) for access token retrieval. To learn how to generate XSUAA OAuth credentials, see: [Retrieve Credentials for Remote Applications](#)

## Context

In simple terms, XS Advanced is basically the Cloud Foundry open-source PaaS with a number of tweaks and extensions provided by SAP. These SAP enhancements include integration with the SAP HANA database, OData support, compatibility with XS classic model, and some additional features designed to improve application security. XS Advanced also provides support for business applications that are composed of multiple micro-services, also known as multi-target applications.

SAP BTP XS Advanced UAA is responsible for the connection of identity providers with business users (for applications). SAP BTP XS Advanced UAA provides authorizations on application level: [role collections](#), [roles](#), [attributes](#), and [role templates](#). To learn more, see: [What Is the SAP Authorization and Trust Management Service?](#)

Follow the steps below to create SAP BTP XS Advanced UAA as a target system to provision SAP BTP users and groups to your Cloud Foundry applications.

These target systems consume SCIM 1.1 API provided by SAP HANA XS Advanced UAA.

## → Remember

You can write users and groups to SAP BTP XS Advanced UAA on an **application** level only. You cannot provision and manage them on a [subaccount](#) level.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP BTP Advanced UAA (Cloud Foundry)* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL to the SCIM API of your SAP BTP XS Advanced UAA system.</p> <p>If not sure about the exact URL, ask your SAP BTP XS UAA administrator.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
OAuth2TokenServiceURL	<p>As you need to make OAuth authentication to the SAP BTP XS Advanced UAA system, enter the URL to the OAuth2 token service</p> <p>If not sure about the exact URL, ask your SAP BTP XS UAA administrator.</p>

Property Name	Description & Value
User	Enter the OAuth client ID of the SAP BTP XS Advanced UAA technical user (see <b>Prerequisites</b> ).
Password	(Credential) Enter the OAuth client secret of the technical user (see <b>Prerequisites</b> ).
xsuaa.origin	<p>Enter the location of your identity provider. To do this:</p> <ol style="list-style-type: none"> <li>1. Open your SAP BTP cockpit.</li> <li>2. Go to your Cloud Foundry global account and choose your subaccount.</li> <li>3. From the left-side navigation, choose <i>Trust Configuration</i>.</li> <li>4. Copy/paste the <i>Origin Key</i> value.</li> </ol> <p>This value will be used as the <i>origin</i> attribute in the system transformation.</p> <p>For more information, see <a href="#">Configure Single and Multiple Origins [page 782]</a></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

Type=*HTTP*

Authentication=*BasicAuthentication*

ProxyType=*Internet*

URL=*https://api.authentication.eu10.hana.ondemand.com*

OAuth2TokenServiceURL=*https://myaccount.authentication.eu10.hana.ondemand.com/oauth/token*

User=*MyXSUAUser*

Password=*\*\*\*\*\**

xsuaa.origin=*myaccount-xsuaa.accounts.ondemand.com*

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP BTP Advanced UAA* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

- **Mapping logic** – The behavior of the default transformation logic is to map all attributes from the intermediate Identity Provisioning representation to the SAP BTP XS Advanced UAA target attributes.
- **User offboarding** – If a user has been deleted from the source system, this change is recognized and the user is deleted in the SAP BTP XS Advanced UAA target system too.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP BTP XS Advanced UAA system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Cloud Foundry UAA API: Users](#) ➦

[Cloud Foundry UAA API: Groups](#) ➦

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.emails.length() > 0) &&
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "constant": "xsuaa-dummy-value",
        "targetPath": "$.id",
        "scope": "createEntity"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true,
        "defaultValue": false
      },
      {
        "sourcePath": "$.verified",
        "targetPath": "$.verified",
        "optional": true,
        "defaultValue": false
      },
      {
        "constant": "%xsuaa.origin%",
        "targetPath": "$.origin"
      },
      {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
      }
    ]
  }
}
```

```

    },
    {
      "condition": "$.emails[?(@.primary == true)].value == []",
      "targetPath": "$.emails[0].primary",
      "constant": true
    },
    {
      "constant": "urn:scim:schemas:core:1.0",
      "targetPath": "$.schemas[0]"
    }
  ]
},
"group": {
  "condition": "('%xsuaa.group.prefix%' === 'null') || ($.displayName
=~ /%xsuaa.group.prefix%.*/)",
  "skipOperations": [
    "create",
    "delete"
  ],
  "mappings": [
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "targetVariable": "entityIdTargetSystem",
      "functions": [
        {
          "condition": "('%xsuaa.group.prefix%' !== 'null') && (@ =~ /
%xsuaa.group.prefix%.*/)",
          "function": "replaceFirstString",
          "regex": "%xsuaa.group.prefix%",
          "replacement": ""
        }
      ]
    },
    {
      "sourceVariable": "entityIdTargetSystem",
      "targetPath": "$.id"
    },
    {
      "sourcePath": "$.description",
      "targetPath": "$.description",
      "optional": true
    },
    {
      "constant": "urn:scim:schemas:core:1.0",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.members",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.members",
      "functions": [
        {
          "function": "resolveEntityIds"
        }
      ]
    },
    {
      "constant": "USER",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.members[*].type",
      "optional": true
    }
  ]
}
}
}

```

5. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[XS CLI: User Administration](#)

[Cloud Foundry UAA: Users](#) ➔

[Cloud Foundry UAA: Groups](#) ➔

### 1.6.2.11.1 Configure Single and Multiple Origins

Configure and provision users with single or multiple origins in SAP BTP XS Advanced UAA (Cloud Foundry) target system.

## Context

An origin tells you which is the identity provider of a user in SAP BTP XS Advanced UAA (Cloud Foundry). It is defined in the trust configuration in the SAP BTP cockpit under ► [Security](#) ► [Trust Configuration](#) ► [Origin Key](#) ►.

The origin itself is not a concept of Identity Provisioning. The role of the service is to ensure that you can read and provision users with their identity providers. Once you find the identity provider in the Origin Key, you need to set it in the `xsuaa.origin` property. You can configure it in source, target and proxy SAP BTP XS Advanced UAA (Cloud Foundry) systems. Both single and multiple values are supported. The value is a string that usually specifies the name of the identity provider or its location.

For example: `xsuaa.origin=ldap` and `xsuaa.origin=ldap;myaccount-xsuaa.accounts.ondemand.com`, where the ";" (semicolon) is the only supported delimiter.

## Provisioning Users with Single Origin

You want to provision a user from a given source system to SAP BTP XS Advanced UAA (Cloud Foundry) target system where a single origin is configured.

1. On the [Properties](#) tab of the target system, enter the value for the `xsuaa.origin` property, for example: [idp1](#).
2. (Optional) Ensure the property resolving the uniqueness of the attributes is configured. By default, it is set to: `xsuaa.user.unique.attribute=userName`.
3. Run a provisioning job from the source system.

As a result, users are created with a single origin in the target system.

## Provisioning Users with Multiple Origins

You want to provision a user from a given source system to SAP BTP XS Advanced UAA (Cloud Foundry) target system where multiple origins are configured.

1. On the [Properties](#) tab of the target system, enter multiple values for the `xsuaa.origin` property, for example: [idp1;idp2](#).
2. Ensure the property resolving the uniqueness of the attributes is configured and contains the [origin](#) value. For example, set it to: `xsuaa.user.unique.attribute=userName,origin`.
3. On the [Transformations](#) tab, update the SAP BTP XS Advanced UAA (Cloud Foundry) target system transformation by adding `[]` (square brackets) to ensure that the following constant supports multiple values:

### JSON Text Editor

#### Code Syntax

```
{
  "constant": "%xsuaa.origin%",
  "targetPath": "$.origin"
},
```

### Graphical Editor



## Note

In case you use the `resolveEntityIds` function in your SAP BTP XS Advanced UAA (Cloud Foundry) target system transformation to resolve the value of one group member in the `sourcePath` to exactly one group member in the `targetPath`, this won't work with multiple origin values. You need to change the `targetPath` in the transformation so that the value of one group member is resolved to multiple group members. See the example below.

## JSON Text Editor

## Graphical Editor

Current mapping:

### Code Syntax

```
{
  "sourcePath": "$.members[0].value",
  "preserveArrayWithSingleElement": true,
  "optional": true,
  "targetPath": "$.members[0].value",
  "functions": [
    {
      "function": "resolveEntityIds"
    }
  ]
},
```

Current mapping:

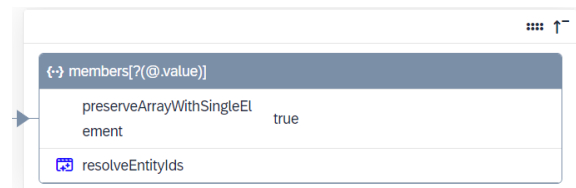


Expected mapping:

### Code Syntax

```
{
  "sourcePath": "$.members[0].value",
  "preserveArrayWithSingleElement": true,
  "optional": true,
  "targetPath": "$.members[?(@.value)]",
  "functions": [
    {
      "function": "resolveEntityIds"
    }
  ]
},
```

Expected mapping:



#### 4. Run a provisioning job.

As a result, two entries for one and the same user are created. Both entries contain the same user attributes (for example, username and email address) and differ only in their origins. The job logs show that one user is read and two users are created.

If the provisioned user is assigned to a group, this group will have two user entries assigned as group members which differ only in their origins.



### **i Note**

Multiple origins are not supported in provisioning scenarios between SAP BTP XS Advanced UAA (Cloud Foundry) source system and SAP BTP XS Advanced UAA (Cloud Foundry) target system.

## **Change From Single to Multiple Origins**

You have provisioned users to SAP BTP XS Advanced UAA (Cloud Foundry) with a single origin and now you want to provision them with multiple ones.

Follow the steps in the *Provisioning Users with Multiple Origins* above. Before you run the provisioning job, you must reset the target system. For more information, see [Reset Identity Provisioning System \[page 1542\]](#)

## **1.6.2.12 SAP Build Work Zone, advanced edition**

Follow this procedure to set up SAP Build Work Zone, advanced edition as a target system.

### **Prerequisites**

You have OAuth credentials for SAP Build Work Zone, advanced edition. To learn how, see [SAP Build Work Zone, advanced edition: Add an OAuth Client](#)

### **Context**

After fulfilling the prerequisites, follow the procedure below to create a target SAP Build Work Zone, advanced edition system to provision users and groups.

These target systems consume SCIM 2.0 API provided by SAP Build Work Zone, advanced edition.

### **Procedure**

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Build Work Zone, advanced edition* as a target system. For more information, see [Add a System \[page 1477\]](#).

- Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	<p>Enter the URL related to your SAP Build Work Zone, advanced edition system, in format: <b>https://&lt;account&gt;&lt;sap_wz_domain&gt;.workzone.ondemand.com</b></p> <p>For example: <i>https://mytenant.mydomain123.workzone.ondemand.com</i></p>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth client key, created for your SAP Build Work Zone, advanced edition tenant (see <b>Prerequisites</b> ).
Password	Enter the OAuth client secret, created for your SAP Build Work Zone, advanced edition tenant (see <b>Prerequisites</b> ).
OAuth2TokenServiceURL	<p>Enter the URL of the access token provider service for your SAP Build Work Zone, advanced edition instance, in format: <b>https://&lt;account&gt;&lt;sap_wz_domain&gt;.workzone.ondemand.com/api/v1/auth/token</b></p> <p>For example: <i>https://myaccount.mydomain123.workzone.ondemand.com/api/v1/auth/token</i></p>
Optional Properties	
(Optional) <code>workzone.content.type</code>	<p>This property makes the SAP Build Work Zone, advanced edition target system to send the specified value for the <i>Content-Type</i> HTTP header.</p> <p>Example: <b>application/json</b></p> <p>Default value (when not specified): <i>application/scim+json</i></p>

Property Name	Description & Value
(Optional) <code>workzone.support.patch.operation</code>	<p>This is a default property – it appears during system creation.</p> <p>Its default value is <i>true</i>. That means, when the Identity Provisioning identifies a changed entity in the source system, it will execute the updates as PATCH requests instead of PUT. That means, only the changes will be written in SAP Build Work Zone, advanced edition, instead of provisioning the whole entity data.</p>
(Optional) <code>workzone.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the SAP Build Work Zone, advanced edition target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s). This property defines by which unique attribute(s) the existing user to be searched (resolved).</p> <p>Default value (when not specified): <i>userName</i></p> <p>To learn more, see: <a href="#">List of Properties</a></p>
(Optional) <code>workzone.group.unique.attribute</code>	<p>If the Identity Provisioning tries to create a group that already exists on the SAP Build Work Zone, advanced edition target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific). To make the search filter by a specific attribute, specify this attribute as a value for this property.</p> <p>Default value (when not specified): <i>displayName</i></p> <p>To learn more, see: <a href="#">List of Properties</a></p>
(Optional) <code>workzone.include.if.match.wildcard.header</code>	<p>This property makes the SAP Build Work Zone, advanced edition target system to send the <i>If-Match</i> HTTP header with a value of "*" for every request to SAP Build Work Zone, advanced edition. This header could be used for entity versioning.</p> <p>Default value (when not specified): <i>false</i></p>
(Optional) <code>ips.failed.request.retry.attempts</code>	Predefined value: <i>2</i>
(Optional) <code>ips.failed.request.retry.attempts.interval</code>	Predefined value: <i>30</i>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Build Work Zone, advanced edition](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Build Work Zone, advanced edition. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Build Work Zone OData API](#) 📄

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id",
        "scope": "deleteEntity"
      },
      {
        // When a user is supposed to be deleted from SAP Work Zone, it actually
        // has its status set to inactive instead of being deleted.
        "constant": false,
        "targetPath": "$.active",
        "scope": "deleteEntity"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": "$.schemas[1]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:sap:2.0:User",
        "targetPath": "$.schemas[2]"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas",
        "optional": true
      },
      {
        "sourcePath": "$.userType",
        "optional": true,
        "targetPath": "$.userType"
      },
      {
        "condition": "$.groups[?(@.value ==
'Workzone_User_Type_public')] EMPTY false",
        "constant": "public",
        "targetPath": "$.userType"
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath": "$.userName",
      "targetPath": "$.userName"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid'],
      "targetPath": "$.userName",
      "optional": true
    },
    {
      "sourcePath": "$.emails",
      "targetPath": "$.emails",
      "preserveArrayWithSingleElement": true,
      "optional": true
    },
    {
      "condition": "$.emails[0].length() > 0",
      "constant": true,
      "targetPath": "$.emails[0].primary"
    },
    {
      "condition": "($.locale EMPTY false) && ($.addresses[?
(@.type == 'work').country EMPTY false)",
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "functions": [
        {
          "function": "toLowerCaseString"
        },
        {
          "function": "concatString",
          "suffix": "_"
        },
        {
          "function": "concatString",
          "suffix": "$.addresses[?(@.type ==
'work')].country"
        }
      ]
    },
    {
      "sourcePath": "$.name",
      "targetPath": "$.name",
      "optional": true
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "optional": true
    },
    {
      "sourcePath": "$.active",
      "targetPath": "$.active",
      "optional": true
    },
    {
      "sourcePath": "$.title",
      "targetPath": "$.title",
      "optional": true
    },
    {
      "sourcePath": "$.timezone",
      "optional": true,
      "targetPath": "$.timezone"
    },
    {

```

```

        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses"
      },
      {
        "condition": "$.addresses[?(@.type == 'work')].country
EMPTY false",
        "constant": true,
        "targetPath": "$.addresses[1].primary"
      },
      {
        "condition": "$.groups[?(@.value == 'Workzone_Admin')]"
EMPTY false",
        "constant": "Administrator",
        "targetPath": "$.roles[0].value"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['value']",

```

```

        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "optional": true,
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['attributes']",
        "targetPath":
"$['urn:sap:cloud:scim:schemas:extension:custom:2.0:JamCustomUser']
['attributes']",
        "optional": true
    }
]
},
"group": {
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath": "$.schemas[0]"
        },
        {
            "sourcePath": "$.schemas",
            "preserveArrayWithSingleElement": true,
            "targetPath": "$.schemas",
            "optional": true
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName"
        },
        {
            "sourcePath": "$.members[*].value",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.members[?(@.value)]",
            "functions": [
                {
                    "type": "resolveEntityIds"
                }
            ]
        }
    ]
}
}

```

5. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe to the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during your jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

### 1.6.2.13 SAP Build Work Zone, standard edition

Follow this procedure to set up SAP Build Work Zone, standard edition as a target system.

## Prerequisites

You have created an instance and generated a service key for the standard service plan of SAP Build Work Zone, standard edition. For more information, see: [Initial Setup](#).

The service key contains the API URL and the OAuth credentials (`clientid` and `clientsecret`) under the `uaa` property.

## Context

The SAP Build Work Zone, standard edition simplifies access to SAP, custom-built, and third party applications and extensions (both on the cloud and on premise) by establishing a central launchpad.

You can use the Identity Provisioning UI to configure SAP Build Work Zone, standard edition as a target system for provisioning users, groups and users' group assignments from various source systems. In SAP Build Work Zone, standard edition, users can only be created and deleted. The update operation is skipped in the default write transformation.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Build Work Zone, standard edition* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.



## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter the SAP Build Work Zone, standard edition API URL from the service key of your SAP Build Work Zone, standard edition instance under endpoints [portal-service]. It follows the pattern:  <code>https://portal-service.cfapps.sap.hana.ondemand.com</code>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client Id, from the service key of your SAP Build Work Zone, standard edition instance under <code>uaa.clientid</code> .
Password	(Credential) Enter the OAuth Client Secret, from the service key of your SAP Build Work Zone, standard edition instance under <code>uaa.clientsecret</code> .
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL from the service key of your SAP Build Work Zone, standard edition instance. It follows the pattern: <code>&lt;uaa.url&gt;/oauth/token</code> .  For example: <code>https://ips-cflp-woaealle.authentication.sap.hana.ondemand.com/oauth/token</code>

Property Name	Value
<code>cflp.providerId</code>	<p>Enter a valid providerID value.</p> <p>The provider ID is specified in the Channel Manager of the SAP Build Work Zone, standard edition when defining a new content provider. For more information about configuring the content provider to use the Identity Provisioning service, see: <a href="#">Manage Content Providers (Cloud)</a>, <a href="#">Manage Content Providers (On Premise)</a>.</p> <div> <p><b>Note</b></p> <p>All users and groups are provisioned to the target SAP Build Work Zone, standard edition system with the providerID defined for this target system. If you want to use different providerIDs, you need to create separate target systems for every providerID.</p> </div>
<code>cflp.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>SAP Build Work Zone, standard edition supports the following unique attributes which are automatically filled in when the target system is added in the service UI:</p> <pre>emails[0].value, [ 'urn:ietf:params:scim:schemas:extension:2.0:mapping' ] [ 'providerId' ],externalId</pre> <ul style="list-style-type: none"> <li>• If the user has an <b>externalId</b>, the conflict is resolved by <b>externalId</b> and <b>providerId</b>.</li> <li>• If the user doesn't have an <b>externalId</b>, the conflict is resolved by <b>email</b> and <b>providerId</b>.</li> </ul> <p>For the conflict to be resolved, an existing user matching both unique attributes should be found. If an existing user doesn't match both unique attributes or matches only one of them, the user creation fails.</p> <div> <p><b>→ Recommendation</b></p> <p>We recommend that you do not modify the value of the <code>cflp.user.unique.attribute</code> property. Otherwise, user creation fails.</p> </div>

Property Name	Value
<code>cflp.group.unique.attribute</code>	<p>If Identity Provisioning tries to provision a group that already exists in the target system (a conflicting group), this property defines the unique attributes by which the existing group will be searched and resolved.</p> <p>SAP Build Work Zone, standard edition supports a pair of unique attributes which is automatically filled in when the target system is added in the service UI:</p> <pre><b>externalId,</b> <b>[ 'urn:ietf:params:scim:schemas:extension:2.0:mapping' ][ 'providerId' ]</b></pre> <p>For the conflict to be resolved, an existing group matching both unique attributes should be found. In this case, Identity Provisioning updates the group. This means, the conflicting group overwrites the existing one. If the group matches only one of the unique attributes, the conflict is not resolved, and the group creation fails.</p> <div> <p>→ Recommendation</p> <p>We recommend that you do not modify the value of the <code>cflp.group.unique.attribute</code> property. Otherwise, the group creation fails.</p> </div>

Property Name	Value
<code>cflp.support.bulk.operation</code>	<p>This property enables bulk operations for users and groups.</p> <p>When the property is enabled (set to <i>true</i>), the following operations can be executed by Identity Provisioning service in a single request:</p> <ul style="list-style-type: none"> <li>• Create or delete multiple users</li> <li>• Create, update, or delete multiple groups</li> </ul> <p>When the property is disabled (set to <i>false</i>), the following operations can be executed by Identity Provisioning service for a single entity at a time:</p> <ul style="list-style-type: none"> <li>• Create or delete a user</li> <li>• Create, update, or delete a group</li> </ul> <div> <p><b>Note</b></p> <p>Update operation is skipped for users in the default write transformation.</p> </div> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i></p>
<code>cflp.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of operations to be provisioned per request.</p> <p><b>Possible values:</b></p> <p>Default value: <i>20</i></p> <p>Maximum value: <i>100</i></p> <p>If you enter a number larger than 100, the service will replace it with the default value (20).</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map user and group attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Build Work Zone, standard edition* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Build Work Zone, standard edition system. For more information, see: [Manage Transformations \[page 1494\]](#).

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target SAP Build Work Zone, standard edition entity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "skipOperations": [
      "update"
    ],
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      },
      {
        "constant": "%cflp.providerId%",
        "targetPath": "$[ 'urn:ietf:params:scim:schemas:extension:2.0:mapping' ][ 'providerId' ]"
      },
      {
        "condition": "$.emails[?(@.primary == true)].value == []",
        "sourcePath": "$.emails[0].value",
        "optional": true,
        "targetPath": "$.emails[0].value"
      },
      {
        "condition": "$.emails[?(@.primary == true)].value != []",
        "sourcePath": "$.emails[?(@.primary == true)].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails[0].value",
        "functions": [
          {
            "function": "elementAt",
            "index": 0
          }
        ]
      },
      {
        "condition": "$.emails[0].length() > 0",
        "constant": true,
        "targetPath": "$.emails[0].primary"
      },
      {
        "sourcePath": "$[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true,
        "targetPath": "$.externalId"
      },
      {
        "sourcePath": "$.groups[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups[?(@.value)]",
        "functions": [
```

```

        {
            "entityType": "group",
            "function": "resolveEntityIds"
        }
    ]
}
]
},
"group": {
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "constant": "%cflp.providerId%",
            "targetPath": "$
['urn:ietf:params:scim:schemas:extension:2.0:mapping']['providerId']"
        },
        {
            "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath": "$.schemas[0]"
        },
        {
            "constant":
"urn:ietf:params:scim:schemas:core:2.0:mapping",
            "optional": true,
            "targetPath": "$.schemas[1]"
        },
        {
            "sourcePath": "$.displayName",
            "optional": true,
            "targetPath": "$.externalId"
        },
        {
            "sourcePath": "$.externalId",
            "optional": true,
            "targetPath": "$.externalId",
            "functions": [
                {
                    "function": "replaceAllString",
                    "regex": "(?i)(^pcd:)",
                    "replacement": ""
                },
                {
                    "function": "replaceString",
                    "target": "/",
                    "replacement": ":"
                },
                {
                    "function": "replaceString",
                    "target": "(",
                    "replacement": "@"
                },
                {
                    "function": "replaceString",
                    "target": ")",
                    "replacement": "+"
                }
            ]
        }
    ]
},
{
    "sourcePath": "$.members[*].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members[?(@.value)]",
    "functions": [

```

```
    "function": "resolveEntityIds"
  }
}
]
```

5. Add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information


[Configure Integration with the Identity Provisioning Service](#)

### 1.6.2.14 SAP Business Network

Follow this procedure to set up SAP Business Network as a target system.

## Context

### i Note

Currently, SAP Business Network connector is only available for selected customers who are approached by SAP. For more information, see [3305074](#) 

SAP Business Network, formerly known as Ariba Network, is a cloud-based offering that makes it possible for buyers and suppliers to collaborate on transactions, strengthen their relationships, and discover new business opportunities.

You can use Identity Provisioning to configure SAP Business Network as a target system where you can provision users and update group members.

## i Note

Identity Provisioning cannot create and delete groups in SAP Business Network target system. It can only update existing groups by adding or removing group members. Therefore, groups must have been created in SAP Business Network before you run a provisioning job.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Business Network* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the SAP Business Network API URL.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth Client Id, created for your SAP Business Network system.
Password	(Credential) Enter the OAuth Client Secret, created for your SAP Business Network system.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.
bn.api.key	An API Key represents the unique key that identifies a particular application as a legitimate consumer of an API.



Property Name	Value
<code>bn.realm.id</code>	The realm name is part of the URL you use to access SAP Business Network.
(Optional) <code>bn.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved. The property is not added automatically at system creation.</p> <p>Default value: <code>userName</code></p> <p>If the service finds an existing user by <code>userName</code>, it updates this user with the data of the conflicting one. If the service does not find an existing user by <code>userName</code>, the creation of the conflicting user fails.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Business Network* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Business Network system. For more information, see:

[Manage Transformations \[page 1494\]](#).

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target SAP Business Network entity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.emails.length() > 0) && ($.name.givenName EMPTY false) && ($.name.familyName EMPTY false)",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": [
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:ietf:params:scim:schemas:extension:sap:2.0:User",

```

```

"urn:ietf:params:scim:schemas:extension:sap.business.network:2.0:User",
  "urn:com.sap.ariba.framework.scim2.common.types.UserResourceV2"
],
"targetPath": "$.schemas"
},
{
  "sourcePath": "$.externalId",
  "targetPath": "$.externalId",
  "optional": true
},
{
  "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
  "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
},
{
  "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sendMail']",
  "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sendMail']",
  "optional": true
},
{
  "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:sap.business.network:2.0:User']
['userStatus']",
  "targetPath":
"$['urn:ietf:params:scim:schemas:extension:sap.business.network:2.0:User']
['userStatus']",
  "optional": true
},
{
  "sourcePath": "$.displayName",
  "targetPath": "$.displayName",
  "optional": true
},
{
  "sourcePath": "$.active",
  "targetPath": "$.active",
  "optional": true,
  "defaultValue": true
},
{
  "sourcePath": "$.name.givenName",
  "targetPath": "$.name.givenName"
},
{
  "sourcePath": "$.name.middleName",
  "targetPath": "$.name.middleName",
  "optional": true
},
{
  "sourcePath": "$.name.familyName",
  "targetPath": "$.name.familyName"
},
{
  "sourcePath": "$.name.formatted",
  "targetPath": "$.name.formattedName",
  "optional": true
},
{
  "sourcePath": "$.name.honorificPrefix",
  "targetPath": "$.name.honorificPrefix",
  "optional": true
},
{

```

```

        "sourcePath": "$.name.honorificSuffix",
        "targetPath": "$.name.honorificSuffix",
        "optional": true
    },
    {
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.emails[0].value"
    },
    {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
    },
    {
        "condition": "$.addresses[?(@.type == 'work')].country EMPTY
false",
        "sourcePath": "$.addresses[?(@.type == 'work')]",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses[0]"
    },
    {
        "sourcePath": "$.phoneNumbers",
        "targetPath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.preferredLanguage",
        "targetPath": "$.preferredLanguage",
        "optional": true
    },
    {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
    },
    {
        "sourcePath": "$.timezone",
        "targetPath": "$.timezone",
        "optional": true
    }
]
},
"group": {
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "constant": ["urn:ietf:params:scim:schemas:core:2.0:Group"],
            "targetPath": "$.schemas"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName"
        },
        {
            "sourcePath": "$.members[*].value",
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": "$.members[?(@.value)]",
            "functions": [
                {
                    "entityType": "group",
                    "type": "resolveEntityIds"
                }
            ]
        }
    ]
}

```

```

    },
    {
      "sourcePath": "$.members[*].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.members[?(@.value)]",
      "functions": [
        {
          "entityType": "user",
          "type": "resolveEntityIds"
        }
      ]
    }
  ]
}

```

5. Add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Business Network](#)

### 1.6.2.15 SAP Central Business Configuration

Follow this procedure to set up SAP Central Business Configuration (in short, CBC) as a target system.

## Prerequisites

You have created a technical user with administrator permissions that will be used to call the API of SAP Central Business Configuration for creating or updating user and group member information.

## Context

Create a CBC target system to provision users and group members to it.

### ⚠ Caution

You can't create or delete groups on CBC. That means:

- On the attempt to create a group on CBC, Identity Provisioning will only add new members or update existing ones. Also, when the service reads a group from a source system, there must be a group with the exact same display name (case sensitive) in the CBC target system. Otherwise, an error will be thrown and the group members will not be updated.
- On the attempt to delete a group on CBC, Identity Provisioning will only remove its members (group assignments). And this can happen only if the relevant group assignments have been provisioned/are present in the target system.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Central Business Configuration* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your CBC system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>

Property Name	Value
User	Specify the technical user for your CBC system.
Password	(Credential) Specify the password for your technical user.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.  For example: <b>https://mycbcaccount.authentication.us2.hana.ondemand.com/oauth/token</b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Central Business Configuration* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your CBC system. For more information, see [Manage Transformations \[page 1494\]](#).

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target CBC entity.

#### Default transformation:

##### Code Syntax

```
{
  "group": {
    "condition": "('%cbc.group.prefix%' === 'null') || ($.displayName
    =~ /%cbc.group.prefix%.*/)",
    "mappings": [
      {
        "targetPath": "$.id",
        "sourceVariable": "entityIdTargetSystem"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath": "$.schema[0]"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "('%cbc.group.prefix%' !== 'null') &&
            (@ =~ /%cbc.group.prefix%.*/)",
            "function": "replaceFirstString",
            "regex": "%cbc.group.prefix%",
            "replacement": ""
          }
        ]
      }
    ]
  },
  {
    "sourcePath": "$.members[*].value",
    "preserveArrayWithSingleElement": true,
```

```

        "optional": true,
        "targetPath": "$.members[?(@.value)]",
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    },
    ],
    },
    "user": {
        "mappings": [
            {
                "targetPath": "$.id",
                "sourceVariable": "entityIdTargetSystem"
            },
            {
                "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
                "targetPath": "$.schemas[0]"
            },
            {
                "constant":
"urn:sap:cloud:scim:schemas:extension:custom:2.0:User",
                "targetPath": "$.schemas[1]"
            },
            {
                "sourcePath": "$.userName",
                "targetPath": "$.userName"
            },
            {
                "sourcePath": "$.name.givenName",
                "targetPath": "$.name.givenName"
            },
            {
                "sourcePath": "$.name.familyName",
                "targetPath": "$.name.familyName"
            },
            {
                "sourcePath": "$.active",
                "targetPath": "$.active"
            },
            {
                "sourcePath": "$.userType",
                "optional": true,
                "targetPath": "$.userType"
            }
        ]
    }
}

```

5. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Central Business Configuration – Collection](#) 

### 1.6.2.16 SAP Commerce Cloud

Follow this procedure to set up SAP Commerce Cloud as a target system.

#### Prerequisites

##### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- In SAP Commerce Cloud, you have added an OAuth client with authorization grant **Client Credentials**. To learn how, see: [Configuring OAuth Client](#).

#### Context

SAP Commerce Cloud is a highly flexible and modular platform for delivering modern customer experiences. It provides a standardized, automated, end-to-end solution that allows your projects to release code from repository to cloud.

You can use Identity Provisioning to configure SAP Commerce Cloud as a target system to provision users and groups. These target systems consume SCIM 2.0 API provided by SAP Commerce Cloud. For more information, see [Commerce Cloud SCIM Web Services API Documentation](#).

#### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Commerce Cloud* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.



## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Enter the URL to your SAP Commerce Cloud system.</p> <p>The URL follows the pattern: <code>https://backoffice.&lt;tenant&gt;.model-t.cc.commerce.ondemand.com</code></p> <p>You can find the correct URL in the <a href="#">Environments</a> section of SAP Cloud Portal.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the client ID to retrieve the OAuth access token for SAP Commerce Cloud.
Password	(Credential) Enter the client secret to retrieve the OAuth access token for SAP Commerce Cloud.
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Commerce Cloud instance.
(Optional) <code>cc.user.unique.attribute</code>	<p>This property defines by which unique attribute(s) an existing user in the target system will be searched and resolved in case Identity Provisioning tries to provision a conflicting user.</p> <p>SAP Commerce Cloud supports the following unique attributes which are automatically filled in during system creation: <code>emails[0].value</code>, <code>userName</code>, <code>externalId</code>.</p> <p>If the service finds an existing user by at least one of the unique attributes, it updates this user with the data of the conflicting one. If such a user is not found, the update of the conflicting user fails. If more than one users with these unique attributes are found, the update fails.</p>

Property Name	Description & Value
(Optional) <code>cc.group.unique.attribute</code>	<p>If you try to provision a group that already exists in a target system, the group creation will fail. In this case, the existing group only needs to be updated.</p> <p>This property defines by which unique attribute(s) the existing group will be searched and resolved. The default value is <b>displayName</b>.</p> <p>If the service finds an existing group by this unique attribute, the group that you try to provision will overwrite the existing one. If such a group is not found, the group that you try to provision will not be created in the target system.</p>
(Optional) <code>cc.support.patch.operation</code>	<p>This property controls how modified entities (users and groups) in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>• If set to <i>true</i>, PATCH operations are used to update users and groups in the target system.</li> <li>• If set to <i>false</i>, PUT operations are used to update users and groups in the target system.</li> </ul> <p>Default value for target systems: <i>false</i></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Commerce Cloud* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

#### **i** Note

Users that are created in SAP Commerce Cloud through the UI, don't have an ID and a username. When synchronizing such users from a source system to SAP Commerce Cloud target system, Identity Provisioning detects that they already exist. To resolve the conflict, the service tries to retrieve them from the target, but as they have no IDs, the users are skipped.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Commerce Cloud system. For more information, see:

[Manage Transformations \[page 1494\]](#).

[Commerce Cloud SCIM Web Services API Documentation](#)

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

## Default transformation:

### Code Syntax

```
{
  "user": {
    "condition": "$.userName EMPTY false",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userId' ]",
        "targetPath": "$.externalId"
      },
      {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.nickName",
        "optional": true,
        "targetPath": "$.nickName"
      },
      {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$.name.middleName",
        "optional": true,
        "targetPath": "$.name.middleName"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.honorificSuffix",
        "optional": true,
        "targetPath": "$.name.honorificSuffix"
      },
      {
        "condition": "($.addresses[*].region EMPTY false) &&
($.addresses[*].country EMPTY false)",
```

```

        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses",
        "functions": [
            {
                "type": "convertCountryRegion",
                "outputFormat": "alpha2",
                "inputAttributes": [
                    "region",
                    "country"
                ],
                "outputAttribute": "region"
            }
        ]
    },
    {
        "sourcePath": "$.userType",
        "optional": true,
        "targetPath": "$.userType"
    },
    {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.active"
    },
    {
        "condition": "$.emails[?(@.primary == true)].value == []",
        "sourcePath": "$.emails[0].value",
        "optional": true,
        "targetPath": "$.emails[0].value"
    },
    {
        "condition": "$.emails[?(@.primary == true)].value != []",
        "sourcePath": "$.emails[?(@.primary == true)].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails[0].value",
        "functions": [
            {
                "function": "elementAt",
                "index": 0
            }
        ]
    },
    {
        "condition": "$.emails[0].length() > 0",
        "constant": true,
        "targetPath": "$.emails[0].primary"
    }
]
},
"group": {
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath": "$.schemas[0]"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName"
        }
    ]
}

```

```

        "condition": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ] EMPTY
false",
        "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
        "targetPath": "$.displayName"
    },
    {
        "sourcePath": "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]",
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    }
]
}
}
}
}
}

```

- Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#).

## Related Information

[SAP Cloud Identity Services Integration Architecture → cloudscimwebservices extension](#)

### 1.6.2.17 SAP Commissions

Follow this procedure to set up a target connector for SAP Commissions.

## Prerequisites

You have created a technical user with administrator permissions that will be used to call the API of SAP Commissions for creating or updating users and user assignments.

## Context

After fulfilling the prerequisites, follow the procedure below to add a target system for SAP Commissions to write users to it. This target system consumes SCIM 2.0 API provided by SAP Commissions.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Commissions* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the SAP Commissions SCIM API portal.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user for your SAP Commissions system.
Password	Enter the password for your SAP Commissions user.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

Type=*HTTP*

Authentication=*BasicAuthentication*

ProxyType=*Internet*

URL=*https://mycommissions.callidus.run/CallidusPortal*

User=*MyCommissionsUser*

Password=*\*\*\*\*\**

4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Commissions](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Commissions. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Commissions REST API](#)

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target entity. If the entity has more than one e-mail addresses, only one of them is used. It is either the e-mail set as primary in the source system, or if no primary e-mail is set, the one that comes first is used.

**Default transformation:**

Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.id",
        "targetVariable": "entityIdTargetSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "optional": true,
        "defaultValue": ""
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true,
        "defaultValue": ""
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName",
        "optional": true,
        "defaultValue": ""
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      }
    ],
    // For each user in SAP Commissions, only one email is written. It can be
    // either the primary email, or the first email from the "emails[]" list.
    {
      "sourcePath": "$.emails[0].value",
      "targetPath": "$.emails[0].value",
      "optional": true,
      "defaultValue": ""
    }
  ]
}
```

```

    {
      "condition": "$.emails[?(@.primary == true)].value != []",
      "sourcePath": "$.emails[?(@.primary == true)].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.emails[0].value",
      "functions": [
        {
          "function": "elementAt",
          "index": 0
        }
      ]
    },
    {
      "constant": "",
      "targetPath": "$.groups[0].value"
    },
    {
      "condition": "$.groups EMPTY false",
      "sourcePath": "$.groups",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.groups",
      "optional": true
    },
    {
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "optional": true,
      "defaultValue": ""
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "optional": true,
      "defaultValue": ""
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
      "targetPath": "$.schemas[0]"
    }
  ]
}

```

5. Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Commissions: Integration with SAP IdP](#)



## 1.6.2.18 SAP Concur

Follow this procedure to set up SAP Concur as a target system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

#### User Provisioning Service (UPS) v4 API with *Pre-2017 Authorization*

- You have created a technical user with administrator permissions that will be used to call the UPS v4 API for creating or updating user account information. For more information, see [User Provisioning Service v4 API](#).
- You have registered a partner application in your SAP Concur system. You need the administrator permissions to register the application. For more information, see [Registering a Partner Application](#).

#### User Provisioning Service (UPS) v4 API or SAP Concur Identity v4 API with *OAuth 2.0* authentication

- You have an SAP Concur admin user with *Web Services Administrator* role assigned.
- Your SAP Concur admin user has obtained a *Company Request Token* and a *Company UUID* from the SAP Concur Company Request Token self-service tool.  
For more information, see [Configure an SAP Concur Entity as an IdP Target](#) → Section 2: SAP Concur Company Request Token.

### Context

Companies that use SAP Concur for managing and controlling travel expenses, invoices and other can use Identity Provisioning service to automate the identity and access management for the SAP Concur solution. You can provision identity data from your existing corporate identity stores, such as SAP AS ABAP or Microsoft Active Directory, or provision employee user data from different SAP cloud user stores, such as SAP SuccessFactors or SAP Cloud Identity Service – Identity Authentication.

SAP Concur offers three types of edition sites: Standard, Professional and Standard-to-Professional Upgrade. Its integration with Identity Provisioning is supported with Professional edition only.

SAP Concur provides two APIs for its integration with Identity Provisioning: UPS v4 API and Identity v4 API (SCIM API). The value of the `concur.api.version` property controls which API you use.

- When the value is set to **1**, or the property is not defined (typical for systems created before versioning was introduced on December 8, 2021), UPS v4 API is used. The UPS v4 API currently supports two authentication methods: *Pre-2017 Authorization* and *OAuth 2.0*. For more information on how to update to version 2, see: [Update Connector Version \[page 1484\]](#)

- When the value is set to **2**, Identity v4 API is used. This is the value that Identity Provisioning automatically sets for newly created systems after versioning was introduced on December 8, 2021. Identity v4 API supports provisioning of users with userUUID attribute which is generated by Identity Authentication at user creation.

To create SAP Concur as a target system, proceed as follows:

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Concur* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note



If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Version 1 Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter: <a href="https://www.concursolutions.com">https://www.concursolutions.com</a>
ProxyType	Enter: <b>Internet</b>
concur.api.version	Defines the version of SAP Concur API. Set the value to <b>1</b> to use UPS v4 API.

Property Name	Description & Value
Authentication	<p>Enter: <a href="#">BasicAuthentication</a></p> <p>When using UPS v4 API (Version 1), two types of <a href="#">BasicAuthentication</a> are supported:</p> <ul style="list-style-type: none"> <li>• <a href="#">Pre-2017 Authorization</a> - Authentication based on Base-64 encoded Concur credentials (LoginID:Password) of the user. For more information, see <a href="#">Pre-2017 Authorization (Deprecated)</a> .</li> <li>• <a href="#">OAuth 2.0</a> - For more information, see <a href="#">Getting Started</a> .</li> </ul>
User	<p>Valid when <a href="#">Pre-2017 Authorization</a> is used.</p> <p>Enter the user ID of the SAP Concur technical user.</p>
Password	<p>Valid when <a href="#">Pre-2017 Authorization</a> is used.</p> <p>(Credential) Enter the password of the SAP Concur technical user.</p>
X-ConsumerKey	<p>Valid when <a href="#">Pre-2017 Authorization</a> is used.</p> <p>(Credential) Enter the key of the registered partner application (see the <b>Prerequisites</b> section).</p>
concur.datacenter	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Specify the SAP Concur data center your Identity Provisioning tenant belongs to. The following SAP Concur data centers are available:</p> <ul style="list-style-type: none"> <li>• us1</li> <li>• us2</li> <li>• eu1</li> <li>• eu2</li> <li>• emea</li> <li>• cn1</li> <li>• usg</li> <li>• int</li> </ul> <p>Based on the provided data center, Identity Provisioning configures the URL of the User Provisioning Service (UPS) v4 API or the SAP Concur Identity v4 API. For example, if you provide <b>us1</b>, the service will configure the URL in the following pattern: <code>us.api.concursolutions.com</code>.</p>

Property Name	Description & Value
<code>concur.authorization.code</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>(Credential) Enter the <a href="#">Company Request Token</a> and run a provisioning job within 24 hours from generating the token in the SAP Concur Company Request Token self-service tool. Otherwise, the token will expire, and you'll need a new one.</p> <p>After the first run of the job, Identity Provisioning fills in automatically a refresh token as the value of the <code>concur.refresh.token</code> property. If a provisioning job has not been run for six months, you'll again need to generate a new token.</p> <div> <p>→ Remember</p> <p>The company request token has a 24 hour validity. If this token expires, you must request a new token.</p> <p>The refresh token has a six month validity. Every time you run a provisioning job, the validity of the refresh token is extended with six months starting from the date of the last run. If you haven't run a provisioning job for six months, your refresh token will expire and you must request a new company request token.</p> </div>
<code>concur.company.id</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Enter the <a href="#">Company UUID</a> as described in the <i>Prerequisites</i> section.</p>
<code>concur.company.domain</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Enter your company domain.</p> <p>The username and the company domain are concatenated in the default transformation in the following format: <code>user@domain</code></p> <p>Your company domain is the part of your username behind the @ symbol. For example: <code>johnsmith@example.com</code></p>

#### Version 2 Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>

Property Name	Description & Value
<code>concur.datacenter</code>	<p>Specify the SAP Concur data center your Identity Provisioning tenant belongs to. The following SAP Concur data centers are available:</p> <ul style="list-style-type: none"> <li>• us1</li> <li>• us2</li> <li>• eu1</li> <li>• eu2</li> <li>• emea</li> <li>• cn1</li> <li>• usg</li> <li>• int</li> </ul> <p>Based on the provided data center, Identity Provisioning configures the URL of the User Provisioning Service (UPS) v4 API or the SAP Concur Identity v4 API. For example, if you provide <b>us1</b>, the service will configure the URL in the following pattern: <code>us.api.concursolutions.com</code>.</p>
<code>concur.api.version</code>	<p>Defines the version of SAP Concur API.</p> <p>Set the value to <b>2</b> to use Identity v4 API. This is the default value of the property.</p>
<code>concur.authorization.code</code>	<p>(Credential) Enter the <a href="#">Company Request Token</a> and run a provisioning job within 24 hours from generating the token in the SAP Concur Company Request Token self-service tool. Otherwise, the token will expire, and you'll need a new one.</p> <p>After the first run of the job, Identity Provisioning fills in automatically a refresh token as the value of the <code>concur.refresh.token</code> property. If a provisioning job has not been run for six months, you'll again need to generate a new token.</p> <div> <p>→ Remember</p> <p>The company request token has a 24 hour validity. If this token expires, you must request a new token.</p> <p>The refresh token has a six month validity. Every time you run a provisioning job, the validity of the refresh token is extended with six months starting from the date of the last run. If you haven't run a provisioning job for six months, your refresh token will expire and you must request a new company request token.</p> </div>

Property Name	Description & Value
<code>concur.company.id</code>	Enter the <a href="#">Company UUID</a> as described in the <i>Prerequisites</i> section.
<code>concur.company.domain</code>	<p>Enter your company domain.</p> <p>The username and the company domain are concatenated in the default transformation in the following format: user@domain</p> <p>Your company domain is the part of your username behind the @ symbol. For example: johnsmith@example.com</p>
<code>concur.user.unique.attribute</code>	<p>This property defines by which unique attribute(s) an existing user in the target system will be searched and resolved in case Identity Provisioning tries to provision a conflicting user.</p> <p>This property appears by default when the system is created. Its value is set to <a href="#">userName</a>.</p> <p>Other possible values:</p> <ul style="list-style-type: none"> <li>• emails[0].value</li> <li>• userName,emails[0].value</li> <li>• Another unique attribute that supports filtering.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Concur](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.


You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Concur. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: Concur \(Users\)](#) 

**UPS v4 API:** [User Provisioning Service v4 API](#) 

**Identity v4 API:** [Identity v4](#) 

- **Mapping logic** – When the SAP Concur system is configured as a target, the default transformation logic offered by the Identity Provisioning service contains the minimum of required properties for the successful provisioning of the users. You can change the default transformation mapping rules to reflect your current setup of entities in the source system. Before you start extending the default transformation, you have to get familiar with the requirements of the SAP Concur API to avoid inconsistencies. For more information, see [User Provisioning Service v4 API](#) .

- **User off-boarding** – Identity Provisioning service handles the end-to-end lifecycle of the users, including their off-boarding. For some source systems, the deletion of a user or inactive user status is the final step of this lifecycle process. The SAP Concur solution, however, does not allow user accounts to be deleted. The offboarding of SAP Concur user accounts is always performed by setting them as disabled. When a user is deleted or set with status **inactive** in a system configured as a source for user data provisioning to SAP Concur, the user account in SAP Concur will be disabled (the attribute "targetPath": "\$.Active" gets a value "N").

### ⚠ Caution

The SAP Concur API requires an initial password setup for all newly provisioned user accounts. The default transformation offers a statement with an empty string as a value for the password configuration. However, it is ignored in order to prevent from a default setup of a wrong initial password for your systems. While the password statement is ignored, the provisioning will not be working. To enable the provisioning to SAP Concur, you need to perform the following operations:

1. Enable the password statement. To do this, either delete "ignore": true, or set it as "ignore": false.
2. Set a proper statement for the password attribute value ("targetPath": "\$.Password").

(Optional) You can leave the default empty string, or you can use the **randomPassword** function to calculate a random value for the initial password of the newly created SAP Concur accounts. If you choose one of these two options and if you are not using single sign-on solution for SAP Concur, you have to also arrange a password reset support process in your company. This will securely offer an initial password to your corporate users for their newly created SAP Concur accounts. For more information, see [Transformation Expressions \[page 330\]](#) → **Transformation Functions**.

### Default transformation when using UPS v4 API (Version 1):

#### ≡ Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.EmpId"
      },
      /* The first array value of the SCIM attribute emails will be used as an
      e-mail address (EmailAddress) for the user record in SAP Concur. */
      {
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.EmailAddress"
      },
      {
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.LoginId"
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.FirstName"
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.LastName"
      },
      {
        "constant": "N",
        "targetPath": "$.Active"
      }
    ]
  }
}
```

```

    },
    {
      "condition": "$.active == true",
      "constant": "Y",
      "targetPath": "$.Active"
    },
    {
      "constant": "N",
      "targetPath": "$.ExpenseApprover"
    },
    {
      "constant": "N",
      "targetPath": "$.ExpenseUser"
    },
    {
      "constant": "N",
      "targetPath": "$.InvoiceApprover"
    },
    {
      "constant": "N",
      "targetPath": "$.InvoiceUser"
    },
    {
      "constant": "N",
      "targetPath": "$.IsTestEmp"
    },
    {
      "constant": "N",
      "targetPath": "$.TripUser"
    },
  ],
  /* An initial password setup is mandatory for all newly provisioned user
  accounts. To enable the provisioning to SAP Concur,
  enable the statement for the Password attribute and make sure its value is
  not empty. For more information, see the Caution box above. */
  {
    "ignore": true,
    "constant": "",
    "targetPath": "$.Password"
  },
  {
    "constant": "USD",
    "targetPath": "$.CrnKey"
  },
  {
    "sourcePath": "$.addresses[?(@.type == 'home')].country",
    "targetPath": "$.CtryCode"
  },
  {
    "constant": "US",
    "targetPath": "$.Custom21"
  },
  {
    "constant": "DEFAULT",
    "targetPath": "$.LedgerName"
  },
  {
    "constant": "DEFAULT",
    "targetPath": "$.LedgerCode"
  },
  {
    "constant": "en_US",
    "targetPath": "$.LocaleName"
  }
]
}

```



The Identity Provisioning does not provide group resource mapping in the SAP Concur write transformation. Although you cannot provision groups to SAP Concur, you can group the users into organizational units by enhancing your target transformation as follows:

#### Sample Code

```
...
{
  "constant": "<provided by SAP Concur>",
  "targetPath": "$.LedgerCode"
},
{
  "constant": "<obtain from SAP Concur API>",
  "targetPath": "$.Custom21"
},
{
  "constant": "<obtain from SAP Concur API>",
  "targetPath": "$.OrgUnit1"
},
{
  "constant": "<obtain from SAP Concur API>",
  "targetPath": "$.OrgUnit2"
}
},
{
  "constant": "DEFAULT" or "<obtain from SAP Concur API>",
  "targetPath": "$.LedgerKey"
},
}
```

#### Default transformation when using Identity v4 API (Version 2):

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant":
[ "urn:ietf:params:scim:schemas:core:2.0:User", "urn:sap:cloud:scim:schemas:extension:custom:2.0:User", "urn:ietf:params:scim:schemas:extension:sap:2.0:User" ],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "functions": [
          {
            "type": "concatString",
            "suffix": "@%concur.company.domain%"
          }
        ]
      }
    ],
    "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
```

```

        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]"
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true,
        "targetPath": "$.externalId"
    },
    {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails",
        "functions": [
            {
                "function": "putIfAbsent",
                "key": "type",
                "defaultValue": "work"
            }
        ]
    },
    {
        "sourcePath": "$.name",
        "targetPath": "$.name"
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName"
    },
    {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
    },
    {
        "sourcePath": "$.timezone",
        "optional": true,
        "targetPath": "$.timezone"
    },
    {
        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses"
    },
    {
        "targetPath": "$.addresses[*].primary",
        "type": "remove"
    },
    {
        "sourcePath": "$.title",
        "optional": true,
        "targetPath": "$.title"
    },
    {
        "sourcePath": "$.phoneNumbers[?(@.type != 'mobile' || (@.type ==
'mobile' && @.primary == false))]",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.phoneNumbers[0]"
    },
    {
        "targetPath": "$.phoneNumbers[*].primary",

```

```

        "type": "remove"
      },
      {
        "sourcePath": "$.phoneNumbers[?(@.type == 'mobile' && @.primary ==
true)]",
        "optional": true,
        "targetPath": "$.phoneNumbers[0]"
      },
      {
        "sourcePath": "$.emergencyContacts",
        "targetPath": "$.emergencyContacts",
        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "constant": "%concur.company.id%",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['companyId']"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['division']"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",

```

```

        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['displayName']"
    }
]
}
}

```

- Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.19 SAP CPQ

Follow this procedure to set up SAP CPQ as a target system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You have created a technical user with administrator permissions that will be used to call the API of SAP CPQ for creating and updating user and group information.
- Make sure all users that need to be read from the source system have an organization unit name. This unit name must correspond to an existing company system ID in SAP CPQ. The *organization unit* and the

[company system ID](#) must be exactly the same. Users with empty (missing) organization units will not be provisioned, as well as users whose organization units don't match any of the SAP CPQ company system IDs for the relevant tenant.

- In order for a created user to be active in SAP CPQ, it should be assigned to an SAP CPQ target group, whose ID ends with suffix [-USERTYPE](#). To learn more, see [SAP CPQ: SCIM API](#) → section **Mappings between SCIM API and SAP CPQ** → groups.

## Context

Create an SAP CPQ target system to provision users and groups to it.

### ⚠ Caution

You can't create or delete groups on SAP CPQ. That means:

- On the attempt to create a group on SAP CPQ, Identity Provisioning will only add new members or update existing ones. Also, if you read a group from a source system, there must be a group with the exact same display name (case sensitive) in the SAP CPQ target system. Otherwise, an error will be thrown and the group members will not be updated.
- On the attempt to delete a group on SAP CPQ, Identity Provisioning will only remove its members (group assignments). And this can happen only if the relevant group assignments have been provisioned/are present in the target system.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add [SAP CPQ](#) as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your SAP CPQ system. It is the same as your SAP CPQ tenant URL. It must contain the domain name.  For example: <b>https://sample1234.mycpqdomain.com</b>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Specify the technical user for your SAP CPQ system. It must also contain the domain name, in format: <code>&lt;user_name&gt;#&lt;domain_name&gt;</code>  For example: <b>JohnSmith#MYCPQDOMAIN</b>
Password	(Credential) Specify the password for your technical user.
<code>ips.delete.existedbefore.entities</code>	Enter: <a href="#">true</a>  SAP CPQ API does not allow creation and deletion of groups. Thus, if a previously read group is later deleted from the source system, this property will delete the relevant members from SAP CPQ (not the group itself).

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP CPQ](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target SAP CPQ entity. You can change the default transformation mapping rules to reflect your current setup of entities in your SAP CPQ. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP CPQ: SCIM API](#)

### Default transformation:

#### Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY false) &&
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "targetPath": "$.id",
```

```

        "sourceVariable": "entityIdTargetSystem"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": "$.schemas[1]"
      },
      {
        "sourcePath": "$.externalId",
        "optional": true,
        "targetPath": "$.externalId"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses"
      },
      {
        "sourcePath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.phoneNumbers"
      },
      {
        "scope": "createEntity",
        "constant": [],
        "targetPath": "$.groups"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']"
      },
      {
        "constant": true,
        "targetPath": "$
['urn:sap:cpq:scim:schemas:extension:custom:2.0:User']['IsSsoUser']"
      },
      {

```

```

        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]"
    }
  ],
  "group": {
    "condition": "('%cpq.group.prefix%' === 'null') || ($.displayName
=~ /%cpq.group.prefix%.*/)",
    "mappings": [
      {
        "targetPath": "$.id",
        "sourceVariable": "entityIdTargetSystem"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath": "$.schemas[0]"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "('%cpq.group.prefix%' !== 'null') &&
(@ =~ /%cpq.group.prefix%.*/)",
            "function": "replaceFirstString",
            "regex": "%cpq.group.prefix%",
            "replacement": ""
          }
        ]
      }
    ]
  },
  {
    "sourcePath": "$.members[*].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members[?(@.value)]",
    "functions": [
      {
        "function": "resolveEntityIds"
      }
    ]
  }
]
}
}

```

- Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).



## 1.6.2.20 SAP Data Custodian

Follow this procedure to set up SAP Data Custodian as a target system.

### Prerequisites


#### ! Restriction

This system is available for all standalone tenants and bundle tenants running on SAP Cloud Identity Services infrastructure. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You have created an SAP Data Custodian tenant.
- You have created a user within your tenant with the required roles for your scenario.
- You have added your user to SAP Identity Service Management (SAP ISM).
- You have completed the Transparency and Control Service Onboarding Process or the Key Management Service Onboarding Process, depending on the scenario you want to implement. For more information, see [Transparency and Control Service Onboarding Process](#) and [Key Management Service Onboarding Process](#).

### Context

#### i Note

Currently, SAP Data Custodian connector is only available for selected customers who are approached by SAP. For more information, see [3319946](#) .

SAP Data Custodian is a robust Software as a Service (SaaS) solution that protects sensitive data stored in public, private, hybrid, and multicloud environments. This solution integrates with partnered public hyperscalers, SAP applications, and SAP managed clouds.

After fulfilling the prerequisites, follow the procedure below to create a target SAP Data Custodian system to provision users and groups.

These target systems consume SCIM 2.0 API provided by SAP Data Custodian.

SCIM API 2.0 does not support managing of group assignments via the SCIM user resource. The "groups" attribute of the user is read-only. This means that the user group assignments should be managed via the SCIM group resources using the "members" attribute (as it is defined by the SCIM standard).

### Procedure

1. Access the Identity Provisioning UI.

- [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Data Custodian* as a target system. For more information, see [Add a System \[page 1477\]](#).
  3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the SAP Data Custodian SCIM API portal.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth client key, created for your SAP Data Custodian tenant.
Password	(Credential) Enter the OAuth client secret, created for your SAP Data Custodian tenant.
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Data Custodian instance, in format: <b>https://&lt;SAP_Data_Custodian_datacenter&gt;/api/v1/auth/token</b>
dc.group.prefix	<p>This property distinguishes SAP Data Custodian groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>DC_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the target system</b>, only groups containing the <b>DC_</b> prefix in their display name will be provisioned to SAP Data Custodian. Groups without this prefix in the display name won't be provisioned.</p> <p>If the property is not set, all groups will be provisioned to SAP Data Custodian.</p>

Property Name	Description & Value
<code>dc.group.unique.attribute</code>	<p>If the service tries to create a group that already exists in the target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific).</p> <p>To make the search filter by a specific attribute, specify this attribute as a value for the <code>dc.group.unique.attribute</code> property.</p> <p>If the property is not specified, the search is done by the default attribute: <i>displayName</i></p>
<code>dc.user.unique.attribute</code>	<p>When Identity Provisioning attempts to provision a user for the first time, it may detect that this user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user will be searched and resolved. If the service finds a user on the target system via this filter, then the conflicting user will overwrite the existing one. If the service does not find a user on the target system via this filter, the creation will fail.</p> <p><b>Default behavior:</b> The property is automatically added during system creation. Its default value is <i>userName</i>. This means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such <i>userName</i> is not found, the creation of the conflicting user fails.</p> <p><b>Possible values:</b></p> <p>Default value: <i>userName</i></p>

Property Name	Description & Value
<code>dc.support.patch.operation</code>	<p>This property controls how modified users in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>• If set to <i>true</i>, PATCH operations are used to update users in the target system. This means, for example, that if a user attribute is modified, only this change will be provisioned and applied in the target system.</li> <li>• If set to <i>false</i>, PUT operations are used to update users in the target system. This means, for example, that if a user attribute is modified, all user attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p>Users can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>• In the source system, some user attributes are modified, or new attributes are added.</li> <li>• In the source system, a condition or a filter is set for users not to be read anymore.</li> <li>• A user is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <code>deleteEntity</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Data Custodian* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Jam Collaboration. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Data Custodian SCIM 2.0 API](#) 

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY false)",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": [
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
          "urn:ietf:params:scim:schemas:extension:sap:2.0:User",
          "urn:sap:cloud:scim:schemas:extension:custom:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "condition": "$.externalId EMPTY false",
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId"
      },
      {
        "condition": "$[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ][ 'userId' ] EMPTY false",
        "sourcePath": "$[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ][ 'userId' ]",
        "targetPath": "$.externalId"
      },
      {
        "condition": "$.emails[?(@.primary == true)].value == []",
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.userName"
      },
      {
        "condition": "$.emails[?(@.primary == true)].value != []",
        "sourcePath": "$.emails[?(@.primary == true)].value",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.userName",
        "functions": [
          {
            "function": "elementAt",
            "index": 0
          }
        ]
      },
      {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.name.familyName"
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath": "$.name.formatted",
      "optional": true,
      "targetPath": "$.name.formatted"
    },
    {
      "sourcePath": "$.name.honorificPrefix",
      "optional": true,
      "targetPath": "$.name.honorificPrefix"
    },
    {
      "sourcePath": "$.name.honorificSuffix",
      "optional": true,
      "targetPath": "$.name.honorificSuffix"
    },
    {
      "sourcePath": "$.name.middleName",
      "optional": true,
      "targetPath": "$.name.middleName"
    },
    {
      "sourcePath": "$.active",
      "optional": true,
      "targetPath": "$.active"
    },
    {
      "sourcePath": "$.emails",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.emails"
    }
  ]
},
"group": {
  "condition": "('%dc.group.prefix' === 'null') || ($.displayName
=~ /%dc.group.prefix.%/)",
  "mappings": [
    {
      "sourceVariable": "entityIdTargetSystem",
      "targetPath": "$.id"
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "functions": [
        {
          "condition": "('%dc.group.prefix' !== 'null') &&
(@ =~ /%dc.group.prefix.%/)",
          "function": "replaceFirstString",
          "regex": "%dc.group.prefix%",
          "replacement": ""
        }
      ]
    },
    {
      "sourcePath": "$.members[*].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.members[?(@.value)]",
      "functions": [
        {
          "function": "resolveEntityIds"
        }
      ]
    }
  ]
}

```

```

    }
  ]
}

```

- Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

### **i Note**

When Identity Authentication is configured as a source system, the default transformation logic:

- Provisions only groups and user group assignments if they are part of the following predefined group list:
  - [SAP\\_Data\\_Custodian\\_Auditor](#)
  - [SAP\\_Data\\_Custodian\\_Service\\_Admin](#)
  - [SAP\\_Data\\_Custodian\\_Key\\_Admin](#)
  - [SAP\\_Data\\_Custodian\\_Key\\_User](#)
- Skips some of the attributes from the identity records.
- Sets primary email for `userName` of the user.

This way, the transformation logic ensures that the identity data, sent to the Identity Provisioning SCIM REST API, is consistent.

## **Next Steps**

- Before starting a provisioning job, you can first subscribe to the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during your jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## **Related Information**

[SAP Data Custodian](#)

## 1.6.2.21 SAP Document Center

Follow this procedure to set up SAP Document Center as a target system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You have an SAP Business Technology Platform user with administration rights for the tenant.
- You have enabled the SAP Document Center service in the cockpit.

### Context

SAP Document Center offers programs (apps) that can be downloaded and run on multiple independent devices. For more information, see [SAP Document Center](#).

It plays the role of a content service for your SAP Business Technology Platform subaccount. To use it as a target system for writing users, follow the procedure below.

### Procedure

1. Assign your SAP Business Technology Platform user **admin** rights for SAP Document Center. To do this, open the [SAP Document Center](#) service tile (in the cockpit), open link [Assign Roles & Set Destinations](#), choose [Administrator](#), and then – [Assign](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add [SAP Document Center](#) as a target system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the [Properties](#) tab to configure the connection settings for your system.

#### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.



If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <b>HTTP</b>
URL	Enter the URL, generated in the cockpit for your subaccount in the <a href="#">SAP Document Center</a> tile. You can take this URL from the <a href="#">Configure SAP Document Center</a> link.  Remove the last slash after ".../admin".
ProxyType	Enter: <b>Internet</b>
Authentication	Enter: <b>BasicAuthentication</b>
User	Enter your SAP Business Technology Platform user (with administrator rights).
Password	Enter the password for your SAP Business Technology Platform user.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Document Center](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Document Center. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Document Center: REST API](#)

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "$.userName EMPTY false",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,

```

```

    "targetPath": "$.firstName"
  },
  {
    "sourcePath": "$.name.familyName",
    "optional": true,
    "targetPath": "$.lastName"
  },
  {
    "sourcePath": "$.emails[0].value",
    "optional": true,
    "targetPath": "$.email"
  },
  {
    "sourcePath": "$.userName",
    "targetPath": "$.loginId"
  }
]
}

```

6. Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.22 SAP Fieldglass

Follow this procedure to set up a target connector for SAP Fieldglass.

## Prerequisites

You have created an API application key and a web service. To do that, follow the steps on page: [Create API Application Key or Web Service](#) and [Web Services Setup](#)

You will need the values of *Virtual Person Name (Username)* and *License Key* for the configuration of your target system (**step 3** below).

## Context

After fulfilling the prerequisites, follow the procedure below to add a target system for SAP Fieldglass to write users to it. This target system consumes SCIM 2.0 API provided by SAP Fieldglass.

## i Note

The system transformation supports groups but cannot directly create new groups. However, you can map groups from the source system to groups from SAP Fieldglass by display name.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Fieldglass* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify your SAP Fieldglass environment URL. For example: <i>https://abc123.fgvms.com</i>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter your <i>Virtual Person Name (Username)</i> – see the <b>Prerequisites</b> section above.
Password	(Credential) Enter your <i>License Key</i> – see the <b>Prerequisites</b> section above.

Property Name	Value
OAuth2TokenServiceURL	<p>Enter your OAuth token URL in the following format:</p> <p><code>https://&lt;Environment_URL&gt;/api/oauth2/v2.0/token</code></p> <p>For example: <a href="https://abc123.fgvms.com/api/oauth2/v2.0/token">https://abc123.fgvms.com/api/oauth2/v2.0/token</a></p>
(Optional) fg.group.prefix	<p>This property distinguishes SAP Fieldglass groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>FG_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Fieldglass source system and will be provisioned to the target system with the following name pattern: <b>FG_&lt;GroupDisplayName&gt;</b>. This way SAP Fieldglass groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Fieldglass groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>FG_</b> prefix in their display name will be provisioned to SAP Fieldglass. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Fieldglass.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Fieldglass* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Fieldglass. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: SAP Fieldglass](#) 

- **Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target entity. If the entity has e-mail addresses, the first entry will be marked as primary.
- **User off-boarding** – Users can be deleted from the target system. Depending on the implementation, this could be done through a user interface (if such exists) or the SCIM REST API. Users could be deactivated, depending on the SAP Fieldglass system implementation. The SCIM core schema defines an attribute **“active”**, whose definition depends on the service provider. For more information, see [SCIM: Singular Attributes](#) ➔

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "scope": "createEntity",
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.name.honorificPrefix",
        "targetPath": "$.name.honorificPrefix",
        "condition": "$.name.honorificPrefix IN ['Mr.', 'Mrs.', 'Ms.',
'Dr.']",
        "optional": true
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.title",
        "targetPath": "$.title",
        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement": true
      },
      {
        "sourcePath": "$.emails[0].value",
```

```

        "targetPath": "$.emails[0].value"
      },
      {
        "optional": true,
        "defaultValue": "work",
        "sourcePath": "$.emails[0].type",
        "targetPath": "$.emails[0].type"
      },
      {
        "defaultValue": true,
        "optional": true,
        "sourcePath": "$.emails[0].primary",
        "targetPath": "$.emails[0].primary"
      },
      {
        "sourcePath": "$.timezone",
        "optional": true,
        "targetPath": "$.timezone"
      },
      {
        "sourcePath": "$.addresses",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.addresses"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": "$.schemas[1]"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
      },
      {
        "scope": "createEntity",
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",

```

```

        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true
    },
    {
        "scope": "createEntity",
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['value']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['displayName']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']
['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['userId']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['userId']",
        "optional": true
    }
]
},
// You cannot create groups but instead, you can map groups from the
// source system to groups from SAP Fieldglass by display name.
// The group deletion operation is skipped.
"group": {
    "skipOperations": [
        "delete"
    ],
    "condition": "('%fg.group.prefix%' === 'null') || ($.displayName =~ /
%fg.group.prefix%.*/)",
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName",
            "functions": [
                {
                    "condition": "('%fg.group.prefix%' !== 'null') && (@ =~ /
%fg.group.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%fg.group.prefix%",
                    "replacement": ""
                }
            ]
        }
    ]
}
]

```

```

    },
    {
      "optional": true,
      "preserveArrayWithSingleElement": true,
      "sourcePath": "$.members[*].value",
      "targetPath": "$.members[?(@.value)]",
      "functions": [
        {
          "function": "resolveEntityIds"
        }
      ]
    },
    {
      "sourcePath": "$.schemas",
      "preserveArrayWithSingleElement": true,
      "targetPath": "$.schemas"
    }
  ]
}

```

5. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.23 SAP Field Service Management

Follow this procedure to set up SAP Field Service Management as a target system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

You have OAuth credentials for SAP Field Service Management. For more information, see: [Generating Client ID & Secret](#)



## Context

SAP Field Service Management is a cloud-based solution that is used to resolve customers issues with end-to-end field service management. For example, it helps customers overcome resource limitations, such as having enough skilled technicians in all locations.

You can use the Identity Provisioning user interface (UI) to configure SAP Field Service Management as a target system where you can provision users and group members. When integrated with Identity Provisioning, the cloud solution supports the following group concept:

- A user is created with a default group assigned. A group is mapped to a company (one of the key organizational units in SAP Field Service Management). When using the Identity Provisioning service API, the group is returned as follows:

```
<GroupName>_<CompanyName>
```

- A user can have one group assignment for each company. As in SAP Field Service Management a user can be assigned to multiple companies, this requires a group assignment for each of the companies the user can access. A user cannot be assigned to different groups mapped to one and the same company.

### ⚠ Caution

You cannot create or delete groups in SAP Field Service Management. In this case, you can expect the following behavior:

- If a new group is created in the source system (for example, Identity Authentication) and you run the provisioning job to SAP Field Service Management, the job will fail and no group will be created in the target system.
- If a group exists both in the source system (for example, Identity Authentication) and the target system - SAP Field Service Management, running a provisioning job will only add new members or update existing ones. In this case, groups in the source and target systems must have the same display name (case sensitive). Otherwise, the job will fail, and no group members will be updated.
- If a group exists in the target SAP Field Service Management system and you try to delete it, Identity Provisioning will only remove its group members. In this case, the relevant group members must exist in the target system.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Field Service Management* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your SAP Field Service Management system. It follows the pattern:  <code>https://&lt;cluster&gt;.coresystems.net</code>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client Id, created for your SAP Field Service Management system.
Password	(Credential) Enter the OAuth Client Secret, created for your SAP Field Service Management system.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.  For example: <code>https://&lt;fsm_account&gt;.coresuite.com/api/oauth2/v1/token</code>

Property Name	Value
(Optional) fsm.group.prefix	<p>This property distinguishes SAP Field Service Management groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>FSM_</b></p> <p>You can use the example value or provide your own.</p> <ul style="list-style-type: none"> <li>When <b>set in the source system</b>, the prefix will be prepended to the name of the groups that are read from the SAP Field Service Management source system and will be provisioned to the target system with the following name pattern: <b>FSM_&lt;GroupDisplayName&gt;</b>. This way SAP Field Service Management groups in the target system will be distinguished from groups provisioned from other applications. If the property is not set, the SAP Field Service Management groups will be read and provisioned to the target system with their actual display names.</li> <li>When <b>set in the target system</b>, only groups containing the <b>FSM_</b> prefix in their display name will be provisioned to SAP Field Service Management. Groups without this prefix in the display name won't be provisioned. If the property is not set, all groups will be provisioned to SAP Field Service Management.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Field Service Management* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Field Service Management system. For more information, see:

[Manage Transformations \[page 1494\]](#).

[Field Service Management - SCIM API](#)

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target SAP Field Service Management entity.

**Default transformation:**

## Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY false) && ($.userName EMPTY false)",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": [
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:ietf:params:scim:schemas:extension:sap:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails"
      },
      {
        "sourcePath": "$"
      }
    ],
    [ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ],
    "optional": true,
    "targetPath": "$"
  ],
  [ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]
]
},
"group": {
  "condition": "('%fsm.group.prefix%' == 'null') || ($.displayName
  =~ /%fsm.group.prefix%.*/)",
  "mappings": [
    {
      "sourceVariable": "entityIdTargetSystem",
      "targetPath": "$.id"
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "functions": [
        {
          "condition": "('%fsm.group.prefix%' != 'null') &&
          (@ =~ /%fsm.group.prefix%.*/)",

```

```

        "function": "replaceFirstString",
        "regex": "%fsm.group.prefix%",
        "replacement": ""
      }
    ],
    },
    {
      "sourcePath": "$.members[*].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.members[?(@.value)]",
      "functions": [
        {
          "function": "resolveEntityIds"
        }
      ]
    }
  ]
}

```

5. Add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.24 SAP HANA Database (Beta)

Follow this procedure to set up SAP HANA Database (Beta) as a target system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You have credentials for a tenant in SAP Business Technology Platform. For more information, see: [Accounts](#)
- You have the necessary connection settings to reach an SAP HANA database.

- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need this only when your SAP HANA DB resides in a remote on-premise system, outside your Neo environment. For more information, see [Cloud Connector](#).

### Note

This is a beta feature available on SAP Business Technology Platform. For more information, see: *Enable beta features* in [Change Subaccount Details](#)

## Context

**SAP HANA Database** is a system (connector) in beta state, which allows you to log into remote systems that have SAP HANA installed. Only provisioning of entity type **user** is currently supported by this connector. That includes user assignments to roles and all types of catalog and repository privileges (*schema*, *analytic*, *application*). For more information about SAP HANA privileges, see:

[SAP HANA: GRANT Statement \(Access Control\)](#)


[SAP HANA: Stored Procedures Used to Grant/Revoke Privileges on Activated Repository Objects](#)

When using this connector, what you actually need is to connect to the JDBC SQL port of SAP HANA. Depending on whether this port is visible or hidden, you have the following use cases:

**Case 1** – The JDBC port is directly accessible by the enabled Identity Provisioning NEO account. That mostly happens when it resides in the same Neo environment as your Identity Provisioning service.

**Case 2** – The JDBC port is not directly accessible by your Neo environment. There are two subcases:

- JDBC port of SAP HANA DB is accessible by a system, which is publicly reachable through SSH protocol. You have to configure your [SAP HANA Database \(Beta\)](#) connector so as to open an SSH tunnel to this system. Set the proxy type to **Internet**.
- JDBC port of SAP HANA DB is accessible by a system, which is reachable through SSH protocol only from an internal network. You need to have the Cloud Connector installed in that network and configure it to allow SSH connections from the Identity Provisioning service account. You have to create an SSH tunnel by using TCP protocol connection configuration from the Cloud Connector. When configuring the access control, specify the SSH host and port to reach the system that has access to the JDBC port. Set the proxy type to **OnPremise**.

**Case 3** – SAP HANA DB is installed in the Cloud Foundry environment. You need to enable SSH access on both space and application level. To do this, execute the relevant console commands in the Cloud Foundry command line tool (see: [Cloud Foundry: Accessing Apps with SSH](#) ). The [SAP HANA Database \(Beta\)](#) connector will open an SSH tunnel to a running application container on the Cloud Foundry space. The space configuration of the security groups allows access to the JDBC port of SAP HANA MDC. You need to have the [Space Developer](#) role. Again, there are two subcases:

- Cloud Foundry landscape is publicly accessible through SSH protocol. Set the proxy type to **Internet**.
- Cloud Foundry landscape is accessible through SSH protocol, which is allowed only from a particular network. You need to have the Cloud Connector installed in that network and configure it to allow SSH connections from the Identity Provisioning service account. Set the proxy type to **OnPremise**.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP HANA Database (Beta)* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

Below are listed all available SAP HANA properties. Some of them can be mandatory and others – optional, depending on your scenario.

#### Mandatory Properties

Property Name	Description & Value
ProxyType	This property is applicable if you use an SSH tunnel ( <code>hana.jdbc.access.type=ssh.tunnel cf.app.ssh.tunnel</code> ). Possible values: <ul style="list-style-type: none"><li>• <b>Internet</b> – if the SSH port is visible in your Neo environment</li><li>• <b>OnPremise</b> – if the SSH port is not directly accessible, and you have to use the Cloud Connector. You have to configure TCP protocol connection to the SSH host and port (specify the configuration properties <code>hana.jdbc.ssh.tunnel.host</code> and <code>hana.jdbc.ssh.tunnel.port</code>).</li></ul>
CloudConnectorLocationId	Relevant when the proxy type is <i>OnPremise</i> . Use it only if your SAP Business Technology Platform account uses more than one Cloud Connector.
<code>hana.jdbc.db.user</code>	Name of the SAP HANA Database user
<code>hana.jdbc.db.password</code>	(Credential)
<code>hana.jdbc.db.host</code>	SAP HANA Database host
<code>hana.jdbc.db.port</code>	30015

Property Name	Description & Value
<code>hana.jdbc.access.type</code>	<p>There are three types of SAP HANA access:</p> <ul style="list-style-type: none"> <li>• <b>direct</b> – It requires only <a href="#">hana.jdbc.db.*</a> properties</li> <li>• <b>ssh.tunnel</b> – it requires <a href="#">hana.jdbc.db.*</a> and <a href="#">hana.jdbc.ssh.tunnel.*</a> properties.</li> <li>• <b>cf.app.ssh.tunnel</b> – It requires <a href="#">hana.jdbc.ssh.tunnel.cf.*</a> properties to establish an SSH tunnel to the Cloud Foundry application, from which to access the JDBC SQL port of SAP HANA.</li> </ul>
<code>hana.jdbc.ssh.tunnel.username</code>	The username used for opening the SSH Tunnel
<code>hana.jdbc.ssh.tunnel.host</code>	SSH Tunnel's host
<code>hana.jdbc.ssh.tunnel.port</code>	22
<code>hana.jdbc.ssh.tunnel.auth.type</code>	<p>Supported SSH authentication types:</p> <ul style="list-style-type: none"> <li>• <b>key</b></li> <li>• <b>pwd</b></li> <li>• <b>otp</b></li> <li>• <b>key+otp</b></li> <li>• <b>key+pwd</b></li> <li>• <b>pwd+otp</b></li> <li>• <b>key+pwd+otp</b></li> </ul>
<code>hana.jdbc.ssh.tunnel.cf.api.url</code>	The URL of the Cloud Foundry API.
<code>hana.jdbc.ssh.tunnel.cf.oauth.token.url</code>	The URL of the OAuth token endpoint.
	<div>→ Remember</div> <p>Remove the <code>/oauth/token</code> part at the end of the URL.</p>
<code>hana.jdbc.ssh.tunnel.cf.org</code>	This is the Cloud Foundry organization.
<code>hana.jdbc.ssh.tunnel.cf.space</code>	This is the Cloud Foundry space.
<code>hana.jdbc.ssh.tunnel.cf.app</code>	<p>This is the Cloud Foundry application to which the <a href="#">SAP HANA Database (Beta)</a> system opens an SSH tunnel. For more information, see: <a href="#">Cloud Foundry: Accessing Apps with SSH</a> ➡</p>
<code>hana.jdbc.ssh.tunnel.cf.app.instance</code>	This is the instance number of the Cloud Foundry application.
<code>hana.jdbc.ssh.tunnel.cf.username</code>	This is the Cloud Foundry user. It has the role <b>Developer</b> for the space where the application is deployed.



Property Name	Description & Value
<code>hana.jdbc.ssh.tunnel.cf.password</code>	(Credential) The password for property <code>hana.jdbc.ssh.tunnel.cf.username</code>
<code>hana.jdbc.ssh.tunnel.password</code>	(Credential) Taken into account only if the authentication type includes <b>pwd</b> . That means any of the following: <ul style="list-style-type: none"> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>pwd</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>pwd+otp</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>key+pwd</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>key+pwd+otp</i></code></li> </ul>
<code>hana.jdbc.ssh.tunnel.totp.secret.key</code>	(Credential) Taken into account only if the authentication type includes <b>otp</b> . That means any of the following: <ul style="list-style-type: none"> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>otp</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>key+otp</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>pwd+otp</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>key+pwd+otp</i></code></li> </ul>
<code>hana.jdbc.ssh.tunnel.private.key</code>	(Credential) Taken into account only if the authentication type includes <b>key</b> . That means any of the following: <ul style="list-style-type: none"> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>key</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>key+pwd</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>key+otp</i></code></li> <li><code>hana.jdbc.ssh.tunnel.auth.type = <i>key+pwd+otp</i></code></li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP HANA Database \(Beta\)](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP HANA Database. For more information, see [Manage Transformations \[page 1494\]](#).

#### Default transformation:

## Code Syntax

```
{
  "user": {
    "condition": "$.userName EMPTY false",
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": $.username"
      },
      {
        "targetPath": $.password_option.password",
        "scope": "createEntity",
        "functions": [
          {
            "type": "randomPassword",
            "passwordLength": 24,
            "minimumNumberOfLowercaseLetters": 1,
            "minimumNumberOfUppercaseLetters": 1,
            "minimumNumberOfDigits": 1,
            "minimumNumberOfSpecialSymbols": 0
          }
        ]
      }
    ],
    {
      "ignore": true,
      "constant": true,
      "targetPath":
"$$.password_option.no_force_first_password_change",
      "scope": "createEntity"
    },
    {
      "constant": true,
      "targetPath": $.deactivate",
      "scope": "deleteEntity"
    },
    {
      "sourcePath": "$.userName",
      "targetPath": $.username",
      "scope": "deleteEntity"
    },
    {
      "constant": false,
      "targetPath": $.deactivate"
    },
    {
      "constant": true,
      "targetPath": $.reset_connect_attempts"
    },
    {
      "ignore": true,
      "constant": true,
      "targetPath": $.force_password_change"
    },
    {
      "ignore": true,
      "constant": true,
      "targetPath": $.enable_password_lifetime"
    },
    {
      "ignore": true,
      "constant": true,
      "targetPath": $.disable_client_connect"
    },
    {
      "constant": "NOW",
      "targetPath": $.valid_from"
    }
  }
}
```

```

    },
    {
      "constant": "FOREVER",
      "targetPath": "$.valid_to"
    },
    {
      "ignore": true,
      "constant": "1970-01-01 00:00:00.0",
      "targetPath": "$.valid_from"
    },
    {
      "ignore": true,
      "constant": "1970-01-01 00:00:00.0",
      "targetPath": "$.valid_to"
    },
    {
      "ignore": true,
      "constant": "role",
      "targetPath": "$.catalog_permissions[0].type"
    },
    {
      "ignore": true,
      "constant": "MONITORING",
      "targetPath": "$.catalog_permissions[0].name"
    },
    {
      "ignore": true,
      "constant": "ADMIN",
      "targetPath": "$.catalog_permissions[0].option"
    },
    {
      "ignore": true,
      "constant": "object_privilege",
      "targetPath": "$.catalog_permissions[1].type"
    },
    {
      "ignore": true,
      "constant": "SELECT CDS METADATA",
      "targetPath": "$.catalog_permissions[1].name"
    },
    {
      "ignore": true,
      "constant": "SYS.USERS",
      "targetPath": "$.catalog_permissions[1].on"
    },
    {
      "ignore": true,
      "constant": "role",
      "targetPath": "$.repository_permissions[0].type"
    },
    {
      "ignore": true,
      "constant": "sap.appcore.auth.p::select_ACCESS_VIEWS_BY_USER",
      "targetPath": "$.repository_permissions[0].name"
    },
    {
      "ignore": true,
      "constant": "application_privilege",
      "targetPath": "$.repository_permissions[1].type"
    },
    {
      "ignore": true,
      "constant": "sap.hana.ide::Catalog",
      "targetPath": "$.repository_permissions[1].name"
    },
    {
      "ignore": true,

```

```

        "constant": true,
        "targetPath": "$.repository_permissions[2].revoke"
      },
      {
        "ignore": true,
        "constant": "analytic_privilege",
        "targetPath": "$.repository_permissions[2].type"
      },
      {
        "ignore": true,
        "constant": "_SYS_BI_CP_ALL",
        "targetPath": "$.repository_permissions[2].name"
      }
    ]
  }
}

```

5. Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.25 SAP Integrated Business Planning for Supply Chain

Follow this procedure to set up SAP Integrated Business Planning for Supply Chain (in short, SAP IBP) as a target system.

## Prerequisites

To establish the connection between Identity Provisioning and SAP Integrated Business Planning for Supply Chain, you need to set up the communication (user, system and arrangement) on SAP Integrated Business Planning for Supply Chain. You can do it now (as a prerequisite) or in the process of configuring SAP Integrated Business Planning for Supply Chain as a target system, as described in step 3.

## Context

SAP Integrated Business Planning for Supply Chain is a cloud-based solution that combines sales and operations planning (S&OP), forecasting and demand, response and supply, demand-driven replenishment, and inventory planning.

You can use Identity Provisioning to configure SAP IBP as a target system where you can provision entities from a particular source system. This scenario supports provisioning **users** and **role assignments**. In SAP Integrated Business Planning for Supply Chain, groups correspond to roles, thus group members are user assignments of a role.

### **i** Note

Identity Provisioning cannot create and delete roles in SAP Integrated Business Planning for Supply Chain target system. It can only create, update and delete user assignments of a role. Therefore, roles must have been created in SAP Integrated Business Planning for Supply Chain system before you run a provisioning job.

For example, if you try to create or delete a role in SAP Integrated Business Planning for Supply Chain, Identity Provisioning will only add or remove the user assignments of that role, respectively.




The Identity Provisioning service manages the complete set of **business partners** and their relevant **business users** (employee).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Integrated Business Planning for Supply Chain* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP Integrated Business Planning for Supply Chain and configure your authentication method (basic or certificate-based).

### **i** Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Integrated Business Planning for Supply Chain target system, select the *Certificate* tab and choose  *Generate*  *Download* , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP Integrated Business Planning for Supply Chain backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose [Scenario ID SAP\\_COM\\_0193](#) (SAP Cloud Identity Provisioning Integration).

### i Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.


If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to your SAP IBP system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter your authentication method: <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>

Property Name	Description & Value
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>
ibp.user.unique.attribute	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Integrated Business Planning for Supply Chain target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <a href="#">personExternalID</a>. That means, if the service finds an existing user by a <a href="#">personExternalID</a> , it updates this user with the data of the conflicting one. If a user with such a <a href="#">personExternalID</a> is not found, the creation of the conflicting user fails.</li> <li>Value = <a href="#">emails[0].value</a>. If the service finds an existing user matching both unique attributes <a href="#">email</a> and <a href="#">personExternalID</a>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <a href="#">email</a>, the update of the existing user fails. If a user with such <a href="#">email</a> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><a href="#">personExternalID</a></li> <li><a href="#">emails[0].value</a></li> </ul> <p>Default value: <a href="#">personExternalID</a></p>

Property Name	Description & Value
<code>ibp.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP IBP target system.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the target system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the target system.</li> </ul> <p>See also: <a href="#">Extended Explanation of the *.user.roles.override Properties</a> </p>
(Optional) <code>ibp.roles.prefix</code>	<p>This property distinguishes SAP Integrated Business Planning for Supply Chain roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>IBP_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the target system</b>, only roles containing the <b>IBP_</b> prefix in their role name will be provisioned to SAP Integrated Business Planning for Supply Chain. Roles without this prefix in the role name won't be provisioned.</p> <p>If the property is not set, all roles will be provisioned to SAP Integrated Business Planning for Supply Chain.</p>
(Optional) <code>ibp.support.bulk.operation</code>	<p>Set this property to <b>true</b> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request. For more information, see: <a href="#">APIs for Business User Management</a></p> <p>If not specified, the default value is <b>false</b>.</p>
(Optional) <code>ibp.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p>The default value, if not specified, is <b>20</b>.</p> <p>The maximum value is <b>100</b>. If you enter a number larger than 100, the service will replace it with the default value (20).</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.



Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://my1234567-api.scmibp.ondemand.com
User=MyIBPCloudUser
Password=*****
ips.date.variable.format=yyyy-MM-dd
ibp.user.roles.override = false
ibp.support.bulk.operation = true
ibp.bulk.operations.max.count = 50
```

---

##### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP IBP](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in your SAP IBP. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Integrated Business Planning API: Business User](#)

[SAP Business Accelerator Hub: SAP IBP](#) 

##### Default transformation:

###### Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY false) &&
isValidEmail ($.emails[0].value)",
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.personExternalID"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.personExternalID",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath": "$.personExternalID",
```

```

        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true,
        "targetPath": "$.user.globalUserID"
    },
    {
        "targetPath": "$.personID",
        "sourceVariable": "entityIdTargetSystem"
    },
    {
        "targetPath": "$.markedForArchivingIndicator",
        "constant": "false"
    },
    {
        "type": "valueMapping",
        "sourcePaths": [
            "$.userType"
        ],
        "targetPath": "$.businessPartnerRoleCode",
        "defaultValue": "BUP003",
        "valueMappings": [
            {
                "key": [
                    "Employee"
                ],
                "mappedValue": "BUP003"
            }
        ]
    },
    {
        "scope": "createEntity",
        "targetPath": "$.validityPeriod.startDate",
        "sourceVariable": "currentDate"
    },
    {
        "scope": "createEntity",
        "targetPath": "$.validityPeriod.endDate",
        "constant": "9999-12-31"
    },
    {
        "scope": "createEntity",
        "sourceVariable": "currentDate",
        "targetPath": "$.user.validityPeriod.startDate"
    },
    {
        "scope": "createEntity",
        "constant": "9999-12-31",
        "targetPath": "$.user.validityPeriod.endDate"
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.personalInformation.firstName",
        "optional": true
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.personalInformation.lastName"
    },
    {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.personalInformation.middleName",
        "optional": true
    },
    {
        "sourcePath": "$.name.formatted",

```

```

        "targetPath": "$.personalInformation.personFullName",
        "optional": true
    },
    {
        "sourcePath": $.userName,
        "targetPath": $.user.userName
    },
    {
        "sourcePath": $.locale,
        "targetPath": $.user.logonLanguageCode,
        "optional": true
    },
    {
        "sourcePath": $.emails[0].value,
        "targetPath": $.workplaceInformation.emailAddress
    },
    {
        "condition": $.active == false,
        "targetPath": $.user.lockedIndicator,
        "constant": "true"
    }
]
},
"group": {
    "condition": "('%ibp.roles.prefix%' === 'null') || ($.displayName =~ /%ibp.roles.prefix%.*/)",
    "mappings": [
        {
            "sourcePath": $.displayName,
            "functions": [
                {
                    "condition": "('%ibp.roles.prefix%' !== 'null') && (@ =~ /%ibp.roles.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%ibp.roles.prefix%",
                    "replacement": ""
                }
            ],
            "targetVariable": "entityIdTargetSystem",
            "scope": "createEntity"
        },
        {
            "sourcePath": $.displayName,
            "functions": [
                {
                    "condition": "('%ibp.roles.prefix%' !== 'null') && (@ =~ /%ibp.roles.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%ibp.roles.prefix%",
                    "replacement": ""
                }
            ],
            "targetPath": $.displayName
        },
        {
            "sourcePath": $.members[*].value,
            "preserveArrayWithSingleElement": true,
            "optional": true,
            "targetPath": $.members[?(@.value)],
            "functions": [
                {
                    "function": "resolveEntityIds"
                }
            ]
        }
    ]
}
]
}
}

```

See also: [Extended Explanation of the \\*user.roles.overwrite Properties](#) 

6. Now, add a source system from which to read users and roles. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP Integrated Business Planning](#)

## 1.6.2.26 SAP Jam Collaboration

Follow this procedure to set up SAP Jam Collaboration as a target system.

### Prerequisites

You get OAuth credentials for SAP Jam Collaboration. If your SAP Jam tenant is of "SCIM provisioning" type, an OAuth client is automatically created for it, with the name **SCIM API Client**. To find this client:

1. Go to the SAP Jam Collaboration admin panel.
2. Choose [Integrations](#) [OAuth Clients](#).
3. For *SCIM API Client*, choose [View](#).
4. Save the [Key](#) and [Secret](#) values – you'll need them later while configuring your SAP Jam Collaboration provisioning system.

To learn more, see: [SAP Jam: Add an OAuth Client](#)

### Context

After fulfilling the prerequisites, follow the procedure below to create a target SAP Jam Collaboration system to provision users and groups.

These target systems consume SCIM 2.0 API provided by SAP Jam Collaboration.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Jam Collaboration* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter the URL related to your SAP Jam system, in format: <b><a href="#">https://&lt;SAP_Jam_datacenter&gt;.sapjam.com</a></b> For example: <a href="#">https://jam4.sapjam.com</a>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth client key, created for your SAP Jam tenant (see <b>Prerequisites</b> ).
Password	(Credential) Enter the OAuth client secret, created for your SAP Jam tenant (see <b>Prerequisites</b> ).
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Jam instance, in format: <b><a href="#">https://&lt;SAP_Jam_datacenter&gt;/api/v1/auth/token</a></b> For example: <a href="#">https://jam4.sapjam.com/api/v1/auth/token</a>

Property Name	Description & Value
(Optional) <code>jam.group.prefix</code>	<p>This property distinguishes SAP Jam Collaboration groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SJC_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the target system</b>, only groups containing the <b>SJC_</b> prefix in their display name will be provisioned to SAP Jam Collaboration. Groups without this prefix in the display name won't be provisioned.</p> <p>If the property is not set, all groups will be provisioned to SAP Jam Collaboration.</p>
(Optional) <code>ips.failed.request.retry.attempts</code>	Predefined value: <b>2</b>
(Optional) <code>ips.failed.request.retry.attempts.interval</code>	Predefined value: <b>30</b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Jam Collaboration* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Jam Collaboration. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: SAP Jam Collaboration](#)

- **Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target entity. If the entity has e-mail addresses, the first entry will be marked as primary.
- **User off-boarding:**
  - Users can be deleted from the SAP Jam Collaboration system via the SCIM REST API. For more information, see [SCIM: Deleting Resources](#).
  - Users can be deactivated by setting the value of their `active` attribute to **false**. For more information, see [SCIM: Singular Attributes](#).

**Default transformation:**

≡ Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "targetPath": "$.id",
        "type": "remove"
      },
      {
        "targetPath": "$.externalId",
        "type": "remove"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id",
        "scope": "deleteEntity"
      },
      {
        "condition": "$.emails[0].length() > 0",
        "constant": true,
        "targetPath": "$.emails[0].primary"
      },
      {
        "constant": false,
        "targetPath": "$.active",
        "scope": "deleteEntity"
      },
      {
        "targetPath": "$.locale",
        "type": "remove"
      },
      {
        "condition": "($.locale EMPTY false) && ($.addresses[?
(@.type == 'work')].country EMPTY false)",
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "functions": [
          {
            "function": "toLowerCaseString"
          },
          {
            "function": "concatString",
            "suffix": "_"
          },
          {
            "function": "concatString",
            "suffix": "$.addresses[?(@.type ==
'work')].country"
          }
        ]
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']['manager']
['value']",
        "functions": [
          {

```

```

        "function": "resolveEntityIds"
      }
    ]
  },
  "group": {
    "ignore": true,
    "condition": "('%jam.group.prefix%' === 'null') || ($.displayName
    =~ /%jam.group.prefix%.*/)",
    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "targetPath": "$.id",
        "type": "remove"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "('%jam.group.prefix%' !== 'null') &&
            (@ =~ /%jam.group.prefix%.*/)",
            "function": "replaceFirstString",
            "regex": "%jam.group.prefix%",
            "replacement": ""
          }
        ]
      },
      {
        "targetPath": "$.members",
        "type": "remove"
      },
      {
        "sourcePath": "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]",
        "functions": [
          {
            "type": "resolveEntityIds"
          }
        ]
      }
    ]
  }
}

```

- Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe to the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during your jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).



2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

### ! Restriction

Bear in mind the following limitations for the number of sent requests during a provisioning job:

- The **SAP Jam SCIM API** allows up to 13,000 requests per hour and up to 200 requests per minute.
- The Identity Provisioning service can handle the 200 requests per minute limit. If more requests are sent during the minute, the service will "wait" until it can execute them.

## 1.6.2.27 SAP Market Communication for Utilities

Follow this procedure to set up SAP Market Communication for Utilities as a target system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

To establish the connection between Identity Provisioning and SAP Market Communication for Utilities, you need to set up the communication user in SAP BTP ABAP environment. You can do it now (as a prerequisite) or in the process of configuring SAP Market Communication for Utilities as a target system, as described in step 3.

### Context

The SAP Market Communication for Utilities application is based on SAP BTP ABAP environment. You can use Identity Provisioning to configure SAP Market Communication for Utilities as a target system to provision entities from a given source system.

This scenario supports writing **users** and **assignments**. In SAP Market Communication for Utilities, groups correspond to roles, thus group members are user assignments of a role.

#### i Note

Identity Provisioning cannot create and delete roles in SAP Market Communication for Utilities target system. It can only create, update and delete user assignments of a role. Therefore, roles must have been created in SAP Market Communication for Utilities target system before you run a provisioning job.

For example, if you try to create or delete a role in SAP Market Communication for Utilities, Identity Provisioning will only add or remove the user assignments of that role, respectively.

The Identity Provisioning service manages the complete set of **business partners** and their relevant **business users** (Employee).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Market Communication for Utilities* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP Market Communication for Utilities and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Market Communication for Utilities target system, select the *Certificate* tab and choose ► *Generate* ► *Download* ⌵, as described in [Generate and Manage Certificates for Outbound Connection \[page 1507\]](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP BTP ABAP environment backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose *Scenario ID* SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

For more information, see [Maintain a Communication Arrangement for Inbound Communication](#) 📖

### i Note

The communication scenario *SAP\_COM\_0193* is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the *Properties* tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.


If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the API URL to your SAP Market Communication for Utilities system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter your authentication method: <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>

Property Name	Description & Value
<code>maco.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Market Communication for Utilities target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>• Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <code>personExternalID</code></li> <li>• <code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p>

Property Name	Description & Value
<code>maco.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP Market Communication for Utilities target system.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the target system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the target system.</li> </ul> <p>Default value (if the property is missing during system creation): <i>true</i></p> <p>Default value (if the property appears during system creation): <i>false</i></p> <p>See also: <a href="#">Extended Explanation of the *.user.roles.override Properties</a> </p>
(Optional) <code>maco.roles.prefix</code>	<p>This property distinguishes SAP Market Communication for Utilities roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SMC_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the target system</b>, only roles containing the <b>SMC_</b> prefix in their role name will be provisioned to SAP Market Communication for Utilities. Roles without this prefix in the role name won't be provisioned.</p> <p>If the property is not set, all roles will be provisioned to SAP Market Communication for Utilities.</p>
(Optional) <code>maco.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request. For more information, see: <a href="#">APIs for Business User Management</a></p> <p>If not specified, the default value is <i>false</i>.</p>

Property Name	Description & Value
(Optional) <code>maco.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p>The default value, if not specified, is <b>20</b>.</p> <p>The maximum value is <b>100</b>. If you enter a number larger than 100, the service will replace it with the default value (20).</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://12345-aaaaa-3333.abap.hana.ondemand.com
User=MyMaCoUser
Password=*****
ips.date.variable.format=yyy-MM-dd
maco.user.roles.override=false
maco.support.bulk.operation=true
maco.bulk.operations.max.count=50
```

##### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Market Communication for Utilities](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in SAP Market Communication for Utilities. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

**Default transformation:**

≡ Code Syntax

```
{
  "user": {
```

```

"condition": "($.emails EMPTY false) &&
isValidEmail($.emails[0].value)",
"mappings": [
  {
    "sourcePath": "$.userName",
    "targetPath": "$.personExternalID"
  },
  {
    "sourcePath": "$.externalId",
    "targetPath": "$.personExternalID",
    "optional": true
  },
  {
    "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
    "targetPath": "$.personExternalID",
    "optional": true
  },
  {
    "targetPath": "$.personID",
    "sourceVariable": "entityIdTargetSystem"
  },
  {
    "targetPath": "$.markedForArchivingIndicator",
    "constant": "false"
  },
  {
    "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
    "optional": true,
    "targetPath": "$.user.globalUserID"
  },
  {
    "type": "valueMapping",
    "sourcePaths": [
      "$.userType"
    ],
    "targetPath": "$.businessPartnerRoleCode",
    "defaultValue": "BUP003",
    "valueMappings": [
      {
        "key": [
          "Employee"
        ],
        "mappedValue": "BUP003"
      }
    ]
  },
  {
    "scope": "createEntity",
    "targetPath": "$.validityPeriod.startDate",
    "sourceVariable": "currentDate"
  },
  {
    "scope": "createEntity",
    "targetPath": "$.validityPeriod.endDate",
    "constant": "9999-12-31"
  },
  {
    "scope": "createEntity",
    "sourceVariable": "currentDate",
    "targetPath": "$.user.validityPeriod.startDate"
  },
  {
    "scope": "createEntity",
    "constant": "9999-12-31",
    "targetPath": "$.user.validityPeriod.endDate"
  }
]

```

```

    },
    {
      "sourcePath": "$.name.givenName",
      "targetPath": "$.personalInformation.firstName",
      "optional": true
    },
    {
      "sourcePath": "$.name.familyName",
      "targetPath": "$.personalInformation.lastName"
    },
    {
      "sourcePath": "$.name.middleName",
      "targetPath": "$.personalInformation.middleName",
      "optional": true
    },
    {
      "sourcePath": "$.name.formatted",
      "targetPath": "$.personalInformation.personFullName",
      "optional": true
    },
    {
      "sourcePath": $.userName,
      "targetPath": $.user.userName
    },
    {
      "sourcePath": $.locale,
      "targetPath": $.user.logonLanguageCode",
      "optional": true
    },
    {
      "sourcePath": $.emails[0].value,
      "targetPath": $.workplaceInformation.emailAddress"
    },
    {
      "condition": "$.active == false",
      "targetPath": $.user.lockedIndicator",
      "constant": "true"
    }
  ]
},
"group": {
  "condition": "('%macro.roles.prefix%' === 'null') || ($.displayName =~ /%macro.roles.prefix%.*\/)",
  "mappings": [
    {
      "sourcePath": $.displayName,
      "functions": [
        {
          "condition": "('%macro.roles.prefix%' !== 'null') && (@ =~ /%macro.roles.prefix%.*\/)",
          "function": "replaceFirstString",
          "regex": "%macro.roles.prefix%",
          "replacement": ""
        }
      ],
      "targetVariable": "entityIdTargetSystem",
      "scope": "createEntity"
    },
    {
      "sourcePath": $.displayName,
      "functions": [
        {
          "condition": "('%macro.roles.prefix%' !== 'null') && (@ =~ /%macro.roles.prefix%.*\/)",
          "function": "replaceFirstString",
          "regex": "%macro.roles.prefix%",
          "replacement": ""
        }
      ]
    }
  ]
}

```



```

    ],
    "targetPath": "$.displayName"
  },
  {
    "sourcePath": "$.members[*].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.members[?(@.value)]",
    "functions": [
      {
        "function": "resolveEntityIds"
      }
    ]
  }
]
}
}
}

```

See also: [Extended Explanation of the \\*user.roles.overwrite Properties](#)

6. Now, add a source system from which to read users and roles. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA Cloud Documentation](#)

## 1.6.2.28 SAP Marketing Cloud

Follow this procedure to set up SAP Marketing Cloud as a target system.

## Prerequisites

To establish the connection between Identity Provisioning and SAP Marketing Cloud, you need to set up the communication (user, system and arrangement) on SAP Marketing Cloud. You can do it now (as a prerequisite) or in the process of configuring SAP Marketing Cloud as a target system, as described in step 3.

## Context

You can use SAP Marketing Cloud as a target system to provision entities from a certain source system. This scenario supports writing **users** and **assignments**. In SAP Marketing Cloud, groups correspond to roles, thus group members are user assignments of a role.

### i Note

Identity Provisioning cannot create and delete roles in SAP Marketing Cloud target system. It can only create, update and delete user assignments of a role. Therefore, roles must have been created in SAP Marketing Cloud system before you run a provisioning job.

For example, if you try to create or delete a role in SAP Marketing Cloud, Identity Provisioning will only add or remove the user assignments of that role, respectively.


The Identity Provisioning service manages the complete set of **business partners** and their relevant **business users** (Employee and Contingent Worker).

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Marketing Cloud* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP Marketing Cloud and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Marketing Cloud target system, select the *Certificate* tab and choose  as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP Marketing Cloud backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose [Scenario ID SAP\\_COM\\_0193](#) (SAP Cloud Identity Provisioning Integration).

For more information, see: [User Provisioning \(with Corporate Identity Provider\)](#)

### i Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.


If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to your SAP Marketing Cloud system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter your authentication method: <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>
User	Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.  Enter the <a href="#">User Name</a> from the communication arrangement. <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>

Property Name	Description & Value
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>
marketing.cloud.user.unique.attribute	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Marketing Cloud target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <a href="#">personExternalID</a>. That means, if the service finds an existing user by a <a href="#">personExternalID</a>, it updates this user with the data of the conflicting one. If a user with such a <a href="#">personExternalID</a> is not found, the creation of the conflicting user fails.</li> <li>Value = <a href="#">emails[0].value</a>. If the service finds an existing user matching both unique attributes <a href="#">email</a> and <a href="#">personExternalID</a>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <a href="#">email</a>, the update of the existing user fails. If a user with such <a href="#">email</a> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><a href="#">personExternalID</a></li> <li><a href="#">emails[0].value</a></li> </ul> <p>Default value: <a href="#">personExternalID</a></p>

Property Name	Description & Value
<code>marketing.cloud.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP Marketing Cloud target system.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the target system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the target system.</li> </ul> <p>See also: <a href="#">Extended Explanation of the *.user.roles.override Properties</a> </p>
(Optional) <code>marketing.cloud.roles.prefix</code>	<p>This property distinguishes SAP Marketing Cloud roles by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SMKC_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the target system</b>, only roles containing the <b>SMKC_</b> prefix in their role name will be provisioned to SAP Marketing Cloud. Roles without this prefix in the role name won't be provisioned.</p> <p>If the property is not set, all roles will be provisioned to SAP Marketing Cloud.</p>
(Optional) <code>marketing.cloud.support.bulk.operation</code>	<p>Set this property to <b>true</b> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request. For more information, see: <a href="#">APIs for Business User Management</a></p> <p>If not specified, the default value is <b>false</b>.</p>
(Optional) <code>marketing.cloud.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p>The default value, if not specified, is <b>20</b>.</p> <p>The maximum value is <b>100</b>. If you enter a number larger than 100, the service will replace it with the default value (20).</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP

Authentication=BasicAuthentication

ProxyType=Internet

URL=https://my1234567-api.s4hana.ondemand.com

User=MyMarketingCloudUser

Password=*****

ips.date.variable.format=yyyy-MM-dd

marketing.cloud.user.roles.override = false

marketing.cloud.support.bulk.operation = true

marketing.cloud.bulk.operations.max.count = 50
```

---

##### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Marketing Cloud](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in your SAP Marketing Cloud. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Marketing Cloud API: Business User](#)

##### Default transformation:

###### Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY false) &&
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.personExternalID"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.personExternalID",
        "optional": true
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.user.userName"
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.user.logonLanguageCode",
        "optional": true
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath": "$.emails[0].value",
      "targetPath": "$.workplaceInformation.emailAddress"
    },
    {
      "condition": "$.active == false",
      "targetPath": "$.user.lockedIndicator",
      "constant": "true"
    }
  ]
},
"group": {
  "condition": "('%marketing.cloud.roles.prefix%' === 'null') ||
($.displayName =~ /%marketing.cloud.roles.prefix%.*/)",
  "mappings": [
    {
      "sourcePath": "$.displayName",
      "targetVariable": "entityIdTargetSystem",
      "scope": "createEntity",
      "functions": [
        {
          "condition": "('%marketing.cloud.roles.prefix%' !== 'null') &&
(@ =~ /%marketing.cloud.roles.prefix%.*/)",
          "function": "replaceFirstString",
          "regex": "%marketing.cloud.roles.prefix%",
          "replacement": ""
        }
      ]
    }
  ],
  {
    "sourcePath": "$.displayName",
    "targetPath": "$.displayName",
    "functions": [
      {
        "condition": "('%marketing.cloud.roles.prefix%' !== 'null') &&
(@ =~ /%marketing.cloud.roles.prefix%.*/)",
        "function": "replaceFirstString",
        "regex": "%marketing.cloud.roles.prefix%",
        "replacement": ""
      }
    ]
  }
],
{
  "sourcePath": "$.members[*].value",
  "preserveArrayWithSingleElement": true,
  "optional": true,
  "targetPath": "$.members[?(@.value)]",
  "functions": [
    {
      "function": "resolveEntityIds"
    }
  ]
}
]
}
}

```

See also: [Extended Explanation of the \\*user.roles.overwrite Properties](#)

- Now, add a source system from which to read users and roles. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA Cloud Documentation](#)

### 1.6.2.29 SAP S/4HANA Cloud

Follow this procedure to set up SAP S/4HANA Cloud as a target system.

## Prerequisites

To establish the connection between Identity Provisioning and SAP S/4HANA Cloud, you need to set up the communication (user, system and arrangement) on SAP S/4HANA Cloud. You can do it now (as a prerequisite) or in the process of configuring SAP S/4HANA Cloud as a target system, as described in step 3.

## Context

SAP S/4HANA Cloud is a complete enterprise resource planning (ERP) system with built-in intelligent technologies and advanced analytics.

You can use Identity Provisioning to configure SAP S/4HANA Cloud as a target system where you can provision users and group members from source systems of your choice. In SAP S/4HANA Cloud, groups correspond to business roles, thus group members are user assignments of a business role.

### **i** Note

Identity Provisioning cannot create and delete business roles in SAP S/4HANA Cloud target system. It can only create, update and delete user assignments of a role. Therefore, business roles must have been created in SAP S/4HANA Cloud system before you run a provisioning job.

For example, if you try to create or delete a business role in SAP S/4HANA Cloud, Identity Provisioning will only add or remove the user assignments of that role, respectively.

In scenarios where SAP S/4HANA Cloud is configured as a target system, the integration with a human resource (HR) system is active and cannot be switched off. In this case, the synchronization of business



partners is managed by the HR system and SAP S/4HANA Cloud. Identity Provisioning can only be used for managing business users (login users and their login information, such as date/time preferences) and role assignments.




In SAP S/4HANA Cloud, business partners are the central master data objects that hold the complete person profile in the SAP S/4HANA Cloud. For example, business partners with role `Employee` and `Contingent Worker`. Business users are persons who can log on to SAP S/4HANA Cloud system to complete business tasks.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP S/4HANA Cloud* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP S/4HANA Cloud and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP S/4HANA target system, select the *Certificate* tab and choose  [Generate](#)  [Download](#) , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP S/4HANA backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose *Scenario ID* `SAP_COM_0193` (SAP Cloud Identity Provisioning Integration).

### i Note

The communication scenario *SAP\_COM\_0193* is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

4. Choose the [Properties](#) tab to configure the connection settings for your system.

#### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties


Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Enter the SAP S/4HANA Cloud API URL.</p> <p>You can find the correct URL in the <a href="#">API-URL</a> field of the communication arrangement set up for communication scenario SAP_COM_0193.</p> <p>For example: <code>https://my123456-api.s4hana.ondemand.com</code></p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a></p>

**! Restriction**

Do not use special symbol ',' (comma) as it is not supported.

Property Name	Description & Value
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
s4hana.cloud.api.version	<p>The version of the system API you use.</p> <p>Version <a href="#">1</a> means your SAP S/4HANA Cloud system uses <a href="#">SAP_COM_0193</a> communication arrangement.</p>
s4hana.cloud.hr.switch.active	<p>A default property, whose only possible value is <b>true</b>. That means, HR integration is enabled for your system.</p> <div> <b>⚠ Caution</b> <p>Do not change this value! Otherwise, your provisioning job will fail.</p> </div>
s4hana.cloud.hr.switch.dependent.role.codes	<p>A default property.</p> <p>As a comma-separated value, add the codes of the roles maintained by the HR integration. Make sure these role codes are part of your <a href="#">read</a> and <a href="#">write</a> transformations.</p> <p>By default, the following codes are added to your system: <b>BP003 and BBP005</b>. That means, your HR integration will support <a href="#">employees</a> and <a href="#">contingent workers</a>.</p>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>

Property Name	Description & Value
<code>s4hana.cloud.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP S/4HANA Cloud target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <div> <p><b>Note</b></p> <p>You can configure this property only when the integration between a human resource (HR) system and SAP S/4HANA Cloud target system is <b>OFF</b>. Since the HR integration is active and cannot be switched off for SAP S/4HANA Cloud target systems, configuring the <code>s4hana.cloud.user.unique.attribute</code> property is currently irrelevant.</p> </div> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>personExternalID</code></li> <li><code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p>

Property Name	Description & Value
<code>s4hana.cloud.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP S/4HANA Cloud target system.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the target system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the target system.</li> </ul> <p>See also: <a href="#">Extended Explanation of the *.user.roles.override Properties</a> </p>
(Optional) <code>s4hana.cloud.support.bulk.operation</code>	<p>Set this property to <i>true</i> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request. For more information, see: <a href="#">APIs for Business User Management</a></p> <p>If not specified, the default value is <i>false</i>.</p>
(Optional) <code>s4hana.cloud.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p>The default value, if not specified, is <i>20</i>.</p> <p>The maximum value is <b>100</b>. If you enter a number larger than 100, the service will replace it with the default value (20).</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP

Authentication=BasicAuthentication

ProxyType=Internet

URL=https://my1234567-api.s4hana.ondemand.com

User=MyS4HANAUser

Password=*****

s4hana.cloud.api.version=1

ips.date.variable.format=yyyy-MM-dd

s4hana.cloud.hr.switch.active=true

s4hana.cloud.hr.switch.dependent.role.codes=BUP003,BBP010,BBP005

s4hana.cloud.user.roles.overwrite=false

s4hana.cloud.support.bulk.operation=true

s4hana.cloud.bulk.operations.max.count=50
```

---

5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. Identity Provisioning offers a default transformation for the [SAP S/4HANA Cloud](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in SAP S/4HANA Cloud. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

**Default transformation:**

≡ Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.personExternalID"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.personExternalID",
        "optional": true
      }
    ]
  }
}
```

```

        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true,
        "targetPath": "$.personExternalID"
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true,
        "targetPath": "$.user.globalUserID"
    },
    {
        "targetPath": "$.personID",
        "sourceVariable": "entityIdTargetSystem"
    },
    {
        "targetPath": "$.markedForArchivingIndicator",
        "constant": "false"
    },
    {
        "type": "valueMapping",
        "sourcePaths": [
            "$.userType"
        ],
        "targetPath": "$.businessPartnerRoleCode",
        "defaultValue": "BUP003",
        "valueMappings": [
            {
                "key": [
                    "Employee"
                ],
                "mappedValue": "BUP003"
            },
            {
                "key": [
                    "Contingent Worker"
                ],
                "mappedValue": "BBP005"
            }
        ]
    },
    {
        "scope": "createEntity",
        "sourceVariable": "currentDate",
        "targetPath": "$.user.validityPeriod.startDate"
    },
    {
        "scope": "createEntity",
        "constant": "9999-12-31",
        "targetPath": "$.user.validityPeriod.endDate"
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.personalInformation.firstName",
        "optional": true
    },
    {
        "condition": "%s4hana.cloud.hr.switch.active% != null &&
%s4hana.cloud.hr.switch.active% == true",
        "optional": true,
        "sourcePath": "$.name.familyName",
        "targetPath": "$.personalInformation.lastName"
    },
    {
        "condition": "%s4hana.cloud.hr.switch.active% == null ||
%s4hana.cloud.hr.switch.active% == false",
        "sourcePath": "$.name.familyName",

```

```

        "targetPath": "$.personalInformation.lastName"
      },
      {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.personalInformation.middleName",
        "optional": true
      },
      {
        "sourcePath": "$.name.formatted",
        "targetPath": "$.personalInformation.personFullName",
        "optional": true
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.user.userName"
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.user.logonLanguageCode",
        "optional": true
      },
      {
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.workplaceInformation.emailAddress",
        "optional": true
      },
      {
        "condition": "$.active == false",
        "targetPath": "$.user.lockedIndicator",
        "constant": "true"
      }
    ]
  },
  "group": {
    "condition": "('%s4hana.cloud.roles.prefix%' === 'null') ||
    ($.displayName =~ /%s4hana.cloud.roles.prefix%.*/)",
    "mappings": [
      {
        "sourcePath": "$.displayName",
        "targetVariable": "entityIdTargetSystem",
        "scope": "createEntity",
        "functions": [
          {
            "condition": "('%s4hana.cloud.roles.prefix%' !== 'null') && (@
            =~ /%s4hana.cloud.roles.prefix%.*/)",
            "function": "replaceFirstString",
            "regex": "%s4hana.cloud.roles.prefix%",
            "replacement": ""
          }
        ]
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName",
        "functions": [
          {
            "condition": "('%s4hana.cloud.roles.prefix%' !== 'null') && (@
            =~ /%s4hana.cloud.roles.prefix%.*/)",
            "function": "replaceFirstString",
            "regex": "%s4hana.cloud.roles.prefix%",
            "replacement": ""
          }
        ]
      }
    ]
  },
  {
    "sourcePath": "$.members[*].value",
    "preserveArrayWithSingleElement": true,
    "optional": true,

```



```

        "targetPath": "$.members[?(@.value)]",
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    }
}
]
}
}

```

See also: [nullExtended Explanation of the \\*user.roles.overwrite Properties](#) 

6. Now, add a source system from which to read users and roles. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA Cloud Documentation](#)

### 1.6.2.30 SAP S/4HANA for procurement planning

Follow this procedure to set up SAP S/4HANA for procurement planning as a target system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

You have technical credentials for SAP S/4HANA for procurement planning. See: [Onboarding](#)

## Context

SAP S/4HANA for procurement planning is a cloud-based solution designed to help you plan procurement activities with regard to the time schedule, as well as the investment planning of items based on a central bill of material.

You can use Identity Provisioning to configure SAP S/4HANA for procurement planning as a target system where you can provision **users** from source systems.

### i Note

SAP S/4HANA for procurement planning does not support groups.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP S/4HANA Procurement Planning* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	<p>Specify the URL to the SCIM API of your SAP S/4HANA for procurement planning system without path information.</p> <p>For example: <code>https://procplanning-api.cfapps.eu10.hana.ondemand.com</code></p>

Property Name	Value
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth Client Id, created for your SAP S/4HANA for procurement planning system.
Password	Enter the OAuth Client Secret, created for your SAP S/4HANA for procurement planning system.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.  For example: <b>https://procplansecurity.authentication.eu10.hana.ondemand.com/oauth/token</b>

Property Name	Value
s4hana.pp.user.unique.attribute	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). <b>If the service finds such a user on the target system via this filter, then the conflicting user will overwrite the existing one.</b> If the service does not find such a user, the creation will fail.</p> <p>According to your use case and system type, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> <li>• <i>emails[0].value</i></li> <li>• <i>userName,emails[0].value</i></li> <li>• <i>externalId</i>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>Default value: <i>userName</i></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP S/4HANA for procurement planning* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP S/4HANA for procurement planning system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: SAP S/4HANA for Procurement Planning](#)

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target SAP S/4HANA for procurement planning entity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.emails"
      },
      {
        "condition": "$.emails[0].length() > 0",
        "targetPath": "$.emails[0].primary",
        "constant": true
      },
      {
        "targetPath": "$.id",
        "type": "remove"
      },
      {
        "type": "remove",
        "targetPath": "$"
      }
    ]
  }
}
```

5. Add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA for procurement planning – Product Page](#)

### 1.6.2.31 SAP S/4HANA On-Premise

Follow the procedure to set up SAP S/4HANA on-premise (also valid for SAP S/4HANA Cloud, private edition) as a target system.

## Prerequisites

### Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)
- You have technical credentials (user and password) for SAP S/4HANA on-premise.
- The SAP S/4HANA on-premise system is version **1809** or higher.
- You have configured your SOA Manager to directly call the following Web services:
  - **ManageBusinessUserIn**
  - **QueryBusinessUserIn**

For more information, see: [Setting Up SOA Manager](#).

## Context

You can use Identity Provisioning to configure SAP S/4HANA on-premise as a target system where you can provision **users** from source systems.

This scenario requires that you first provision the users to SAP AS ABAP. The created ABAP users will have PFCG roles assigned. Once this is done, the users will have ABAP IDs. Then, after the provisioning to SAP S/4HANA on-premise, business users will be created and linked to the relevant ABAP IDs via the [User\\_Assignment](#) attribute.

In SAP S/4HANA, a business user is defined as a natural person who is represented by a business partner and a link to a user in the system. Actually, the business user is an AS ABAP (SU01) user who also has a one-to-one relation to a corresponding business partner. For more information on the identity model for business users, see SAP Note [2570961](#).

SAP S/4HANA on-premise supports provisioning of users with [User UUID](#) attribute which is generated by Identity Authentication at user creation. The attribute mapping is handled by the default transformation of AS ABAP connector. For more information, see: [SAP Application Server ABAP \[page 741\]](#).

Provisioning of **roles** (considered as [groups](#)) is also handled by the default transformation of SAP AS ABAP connector.

According to your use case, you can decide whether to use SAP S/4HANA on-premise with HR (human resources) integration active or not.

- System with HR integration – When the HR integration is active, business users in SAP S/4HANA on-premise are automatically created and managed by the HR system (for example, an external data source, such as an identity management system). The Identity Provisioning service can manage only the user-related login information, such as date/time preferences, or role assignments. This means that once the users are provisioned to AS ABAP, you need to run the provisioning to SAP S/4HANA on-premise, so that business users (employee, collaboration user, contingent worker, resource) can be linked to the AS ABAP users via the [User\\_Assignment](#) attribute.

#### Note

Identity Provisioning cannot delete business users in SAP S/4HANA Cloud on-premise system with active HR integration. Deletion is possible only in case of erroneous user assignments to business partners. For more information, see [2568251](#).

- System without HR integration – lean business users will be created after the provisioning job, and the AS ABAP users will be linked to them. The Identity Provisioning service manages the complete set of **business partners** and their relevant **business users**.

To enable HR integration, you need to specify the relevant property in the system configuration. See **step 4** from the procedure below.

## Procedure

1. Open the Cloud Connector to add an access control system mapping for **SAP S/4HANA On-Premise**. This is needed to allow the Identity Provisioning service to access SAP S/4HANA On-Premise as a back-end system on the intranet. To learn how, see: [Configure Access Control \(HTTP\)](#)
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add **SAP S/4HANA On-Premise** as a target system. For more information, see [Add a System \[page 1477\]](#).

- Choose the [Properties](#) tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to your SAP S/4HANA On-Premise system.
ProxyType	Enter: <a href="#">OnPremise</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the technical user for SAP S/4HANA On-Premise.
Password	(Credential) Enter the password for the SAP S/4HANA On-Premise technical user.
<code>s4hana.onprem.hr.switch.active</code>	<p>This property is enabled by default. Possible values:</p> <ul style="list-style-type: none"> <li><b>true</b> (default value) – HR integration is enabled for your system</li> <li><b>false</b> – HR integration is disabled for your system</li> </ul>
<code>s4hana.onprem.hr.switch.dependent.role.codes</code>	<p>A default property. Relevant only for systems with activated HR integration, that is if <code>s4hana.onprem.hr.switch.active = <a href="#">true</a></code>.</p> <p>As a comma-separated value, add the codes of the roles maintained by the HR integration. Make sure these role codes are part of your <a href="#">read</a> and <a href="#">write</a> transformations.</p> <p>By default, the following codes are added to your system: <b>BUP003, BBP005, BUP012, WFM001</b>. That means, your HR integration will support <a href="#">employees</a>, <a href="#">contingent worker</a>, <a href="#">collaboration user</a>, and <a href="#">resource</a>.</p>
<code>ips.date.variable.format</code>	<a href="#">yyyy-MM-dd</a>



Property Name	Description & Value
<code>s4hana.onprem.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP S/4HANA On-Premise target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>personExternalID</code></li> <li><code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p>
(Optional) <code>s4hana.onprem.sap-client</code>	<p>Use this property if you want to specify a particular AS ABAP client to use as the <b>sap-client</b> URL parameter.</p> <p>If this property is not specified, the URL will open your default AS ABAP client. To learn more, see: <a href="#">Specifying the Client</a></p> <p>For more information about <b>sap-client</b>, see: <a href="#">SAP URL Parameters</a></p>
(Optional) <code>s4hana.onprem.support.bulk.operation</code>	<p>Set this property to <code>true</code> if you want to enable bulk operations for provisioning entities. That means, the Identity Provisioning service can write, update, and delete multiple users or groups in a single request. For more information, see: <a href="#">APIs for Business User Management</a></p> <p>If not specified, the default value is <code>false</code>.</p>

Property Name	Description & Value
(Optional) <code>s4hana.onprem.bulk.operations.max.count</code>	<p>If you have enabled the bulk operations, you can use this property to set the number of users to be provisioned per request.</p> <p>The default value, if not specified, is <b>20</b>.</p> <p>The maximum value is <b>100</b>. If you enter a number larger than 100, the service will replace it with the default value (20).</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=OnPremise
URL=http://aaa777.myhost:1234
User=MYS4HANAUSER
Password=*****
ips.date.variable.format=yyyy-MM-dd
s4hana.onprem.hr.switch.active=true
s4hana.onprem.hr.switch.dependent.role.codes=BUPO03,BBP010,BBP005
s4hana.onprem.sap-client=101
s4hana.onprem.support.bulk.operation=true
s4hana.onprem.bulk.operations.max.count=50
```

#### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP S/4HANA On-Premise](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules depending on your setup of entities in SAP S/4HANA On-Premise. For more information, see:

[Manage Transformations \[page 1494\]](#)

[APIs for Business User Management](#)

**Default transformation:**

## Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.personExternalID"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.personExternalID",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath": "$.personExternalID",
        "optional": true
      },
      {
        "targetPath": "$.personID",
        "sourceVariable": "entityIdTargetSystem"
      },
      {
        "type": "valueMapping",
        "sourcePaths": [
          "$.userType"
        ],
        "targetPath": "$.businessPartnerRoleCode",
        "defaultValue": "BUP003",
        "valueMappings": [
          {
            "key": [
              "Employee"
            ],
            "mappedValue": "BUP003"
          },
          {
            "key": [
              "Contingent Worker"
            ],
            "mappedValue": "BBP005"
          },
          {
            "key": [
              "Collaboration User"
            ],
            "mappedValue": "BUP012"
          },
          {
            "key": [
              "Resource"
            ],
            "mappedValue": "WFM001"
          }
        ]
      }
    ],
    "scope": "createEntity",
    "targetPath": "$.validityPeriod.startDate",
    "sourceVariable": "currentDate"
  },
  {
    "scope": "createEntity",
    "targetPath": "$.validityPeriod.endDate",
```

```

        "constant": "9999-12-31"
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.personalInformation.firstName",
        "optional": true
    },
    // The following conditions refer to HR integration for your SAP S/4HANA
    // Cloud system. If HR integration is activated (i.e. property
    // s4hana.cloud.hr.switch.active is set to true), then you don't need to
    // provide family name for the users.
    // If it's deactivated (i.e. property s4hana.cloud.hr.switch.active is
    // missing or set to false,
    // then you have to provide family name. You can apply these conditions to
    // different user attributes, analogically to name.familyName.
    {
        "condition": "%s4hana.onprem.hr.switch.active% != null &&
%s4hana.onprem.hr.switch.active% == true",
        "optional": true,
        "sourcePath": "$.name.familyName",
        "targetPath": "$.personalInformation.lastName"
    },
    {
        "condition": "%s4hana.onprem.hr.switch.active% == null ||
%s4hana.onprem.hr.switch.active% == false",
        "sourcePath": "$.name.familyName",
        "targetPath": "$.personalInformation.lastName"
    },
    {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.personalInformation.middleName",
        "optional": true
    },
    {
        "sourcePath": "$.name.formatted",
        "targetPath": "$.personalInformation.personFullName",
        "optional": true
    },
    {
        "sourcePath": "$.nickName",
        "targetPath": "$.personalInformation.nickName",
        "optional": true
    },
    {
        "sourcePath": "$.userName",
        "targetPath": "$.userAssignment.userID",
        "optional": true,
        "functions": [
            {
                "function": "toUpperCaseString"
            }
        ]
    },
    {
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.workplaceInformation.emailAddress",
        "optional": true
    }
]
}
}

```

6. Now, add a source system from which to read users and roles. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[SAP S/4HANA On-Premise](#)

[APIs for Business User Management](#)

[Maintain Collaboration Users](#)

## 1.6.2.32 SAP Sales Cloud and SAP Service Cloud


Follow this procedure to set up SAP Sales Cloud and SAP Service Cloud, formerly known as [SAP Cloud for Customer](#) (in short, C4C), as a target system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

To integrate SAP Sales Cloud and SAP Service Cloud with Identity Provisioning, you need to use SAP Cloud Integration (SAP CI). This service provides a package with integration flows (iFlows) for enabling the creation of users and assignment of users to groups via SCIM API in SAP Sales Cloud and SAP Service Cloud.

- To configure SAP Cloud Integration and SAP Sales Cloud and SAP Service Cloud, see: [Identity Provisioning in SAP Cloud for Customer using System for Cross-Domain Identity Management \(SCIM\)](#)
- To set up and use the [SAP Cloud for Customer Integration with Identity Provisioning via System for Cross-domain Identity Management](#) package, see: [SAP Business Accelerator Hub: SAP Cloud for Customer Integration with Identity Provisioning via System for Cross-domain Identity Management](#) 

## Context

SAP Sales Cloud and SAP Service Cloud is a cloud-based solution that helps customers manage day-to-day sales and service interactions by sending and receiving signals between front- and back-office solutions and providing a single view of the customer.

You can use Identity Provisioning to configure SAP Sales Cloud and SAP Service Cloud as a target system where you can provision users and group members from source systems. Keep in mind that once you have provisioned the entities to SAP Sales Cloud and SAP Service Cloud, a *business user* and an *employee* are created for every provisioned user. The business user is required for the provisioned user to log into the SAP Sales Cloud and SAP Service Cloud system.

This scenario is relevant for new SAP Sales Cloud and SAP Service Cloud customers (green-field approach). This means that users are first provisioned to Identity Authentication (uploaded from files or provisioned by Identity Provisioning from another source system) and afterwards provisioned to SAP Sales Cloud and SAP Service Cloud using Identity Provisioning.

SAP Sales Cloud and SAP Service Cloud provides three versions of APIs. They differ in type and require configuration of specific set of properties (see **step 3.** in the main **Procedure**). By default, the Identity Provisioning service uses version **3**.

### Version 1 (SOAP-based API)

#### i Note

The SOAP-based API version 1 is deprecated.

When created via API version 1, users are initially transferred to a staging area, and then can be replicated to the C4C system manually or via a job, depending on your tenant setup. This API version is SOAP based. In the SAP Sales Cloud and SAP Service Cloud admin console you can distinguish it by its name – */humancapitalmanagementmasterd6*.

To learn how to replicate users (using API v.1), see: [Employee Master Data Replication](#)

### Version 2 (SOAP-based API)

When using API version 2, users are created immediately – there is no need to transfer them to a staging area. This API version is SOAP based. In the C4C admin console, you can distinguish it by its name */employee replicationin2*.

### Version 3 (SCIM 2.0 based API)

When using API version 3, users are created immediately. There is no need to transfer them to a staging area.

For more information on how to set up and use the *Identity Provisioning in SAP Cloud for Customer using a System for Cross-domain Identity Management (SCIM)* package, see the *Prerequisites* section.

#### i Note

If, for some reason, you have access to all API versions, you must use **separate** provisioning systems for each version, to avoid data inconsistency errors.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP Sales Cloud and SAP Service Cloud* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter SAP Cloud Integration runtime URL.  See: <a href="#">How to Get SAP Cloud Integration Runtime URL</a>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter SAP Cloud Integration user ID to connect to SAP Cloud Integration. See: <ul style="list-style-type: none"><li>• <a href="#">Setting Up Inbound HTTP Connections (with Basic Authentication), Neo Environment</a></li><li>• <a href="#">Basic Authentication of IdP User for Integration Flow Processing (Cloud Foundry environment)</a></li></ul>
Password	(Credential) Enter SAP Cloud Integration password to connect to SAP Cloud Integration. See: <ul style="list-style-type: none"><li>• <a href="#">Setting Up Inbound HTTP Connections (with Basic Authentication), Neo Environment</a></li><li>• <a href="#">Basic Authentication of IdP User for Integration Flow Processing (Cloud Foundry environment)</a></li></ul>

Property Name	Description & Value
c4c.api.version	<p>The version of the SAP Sales Cloud and SAP Service Cloud API you use. Possible values – <b>1</b> (deprecated), <b>2</b>, or <b>3</b>. By default, the Identity Provisioning service uses version <b>3</b>.</p> <div> <p><b>i Note</b></p> <p>After you set up the communication arrangement, you can determine the API version used by your SAP Sales Cloud and SAP Service Cloud system. It represents the ID at the end of your generated URL – the name of API v.1 is <a href="#">humancapitalmanagementmasterd6</a>, and for API v.2 is <a href="#">employeereplicationin2</a>.</p> </div>
<b>Relevant for API v.1 (deprecated)</b>	
RemoteSystemID	Enter the system instance ID, configured for the communication system setting in the C4C system.
<b>Relevant for API v.2</b>	
RecipientPartyID	<p>Enter the recipient system name.</p> <p>Example: <b>0011SAP</b></p>
SenderPartyID	<p>Enter the name of the sender system name. It's equal to the value of property RemoteSystemID from API v.1.</p> <p>For example: <b>IPS</b></p>
(Optional)c4c.group.prefix	<p>This property distinguishes SAP Sales Cloud and SAP Service Cloud groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>C4C_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the target system</b>, only groups containing the <b>C4C_</b> prefix in their display name will be provisioned to SAP Sales Cloud and SAP Service Cloud. Groups without this prefix in the display name won't be provisioned.</p> <p>If the property is not set, the SAP Sales Cloud and SAP Service Cloud groups will be read and provisioned to the target system with their actual display names.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 4. Configure the transformations.



Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Sales Cloud and SAP Service Cloud](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in C4C. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: C4C](#)

[SAP Cloud for Customer OData API v2 Reference](#)

### Using API version 1 (deprecated)

If you use the first version of the SAP Sales Cloud and SAP Service Cloud API (SOAP based) – [humancapitalmanagementmasterd6](#) – your systems will be created with `c4c.api.version=1`. You need to use the transformation below and specify the mandatory attribute `RemoteSystemID`. The following interface is used for replicating employee master data to SAP Sales Cloud and SAP Service Cloud: [Inbound Service HumanCapitalManagementMasterDataReplicationEmployeeMasterDataReplicationIn](#)

Along with replicating employees in SAP Sales Cloud and SAP Service Cloud, a business user is created for every user.

Default transformation:

#### Code Syntax

```
/* Attribute RemoteObjectID stores the user name from the source system
into C4C. */
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.RemoteObjectID"
      },
      /* Statements that start with PersonalDetails are related to the employee
      created in C4C. */
      {
        "sourceVariable": "currentDate",
        "targetPath": "$.PersonalDetails.ValidityPeriod.StartDate",
        "functions": [
          {
            "type": "manipulateDate",
            "targetDateFormat": "yyyy-MM-dd"
          }
        ]
      },
      {
        "constant": "9999-12-31",
        "targetPath": "$.PersonalDetails.ValidityPeriod.EndDate"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.PersonalDetails.GivenName"
      },
      {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.PersonalDetails.FamilyName"
      }
    ]
  }
}
```

```

    },
    /* Statements that start with EmployeeType are supported by the C4C system
    only for internal employees.
    (Service agents are not supported as EmployeeType. The supported employee
    types are mandatory and relevant only to lean employees).
    The value of the currentDate variable (the date when the provisioning is
    executed) is set as validity start date of the employee.
    In the default transformation statement, it's converted to the format
    required by C4C via a transformation function. */
    {
        "sourceVariable": "currentDate",
        "targetPath": "$.EmployeeType.ValidityPeriod.StartDate",
        "functions": [
            {
                "type": "manipulateDate",
                "targetDateFormat": "yyyy-MM-dd"
            }
        ],
        {
            "constant": "9999-12-31",
            "targetPath": "$.EmployeeType.ValidityPeriod.EndDate"
        },
        {
            "sourcePath": "$.userName",
            "targetPath": "$.Identity.ID"
        },
        {
            "constant": "false",
            "targetPath": "$.Identity.UserAccountsInactiveIndicator"
        },
        {
            "sourcePath": "$.phoneNumbers[?(@.type == 'mobile')].value",
            "optional": true,
            "targetPath":
                "$.WorkplaceAddress.MobilePhoneNumberDescription"
        },
        {
            "sourcePath": "$.phoneNumbers[?(@.type == 'work')].value",
            "optional": true,
            "targetPath": "$.WorkplaceAddress.PhoneNumberDescription"
        },
        {
            "sourcePath": "$.emails[0].value",
            "optional": true,
            "targetPath": "$.WorkplaceAddress.EmailURI"
        }
    ]
}

```

## Using API version 2

If you want to use the second C4C API (SOAP based) – [employeeereplicationin2](#) – you have to set [c4c.api.version=2](#), change the transformation with the one below, and specify the two mandatory attributes – [RecipientPartyID](#) and [SenderPartyID](#).

Default transformation:

### Code Syntax

```

{
  "user": {

```

```

        "condition" : "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'].employeeNumb
er EMPTY false",
        "mappings": [
            {
                "targetPath": "$.ReceiverEmployeeID",
                "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'].employeeNumb
er"
            },
            {
                "targetPath": "$.BusinessPartnerID",
                "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'].employeeNumb
er"
            },
            {
                "targetPath": "$.EmployeeType.ValidityPeriod.StartDate",
                "sourceVariable": "currentDate",
                "functions": [
                    {
                        "type": "manipulateDate",
                        "targetDateFormat" : "yyyy-MM-dd"
                    }
                ]
            },
            {
                "targetPath": "$.EmployeeType.ValidityPeriod.EndDate",
                "constant": "9999-12-31"
            },
            {
                "targetPath": "$.Common.Name.GivenName",
                "sourcePath": "$.name.givenName",
                "optional": true
            },
            {
                "targetPath": "$.Common.Name.FamilyName",
                "sourcePath": "$.name.familyName"
            }
        ],
        /* You can set a custom namespace for an attribute. For example, if your
        namespace prefix is called a123,
        enter the following lines in your transformation:
        {
            "targetPath": "$['a123:PersonalDetails']
['a123:FamilyName']",
            "sourcePath": "$.name.familyName"
        }
        When sending the request to C4C, the Identity Provisioning service will
        transform this data into XML elements, as follows:
        <a123:PersonalDetails>
            <a123:FamilyName>...</FamilyName>
        </a123:PersonDetails>    */
        {
            "targetPath": "$.Identity.IdentityID",
            "sourcePath": "$.userName"
        },
        {
            "targetPath": "$.Identity.UserAccountsInactiveIndicator",
            "constant": "false"
        },
        {
            "condition": "$.active == false",
            "targetPath": "$.Identity.UserAccountsInactiveIndicator",
            "constant": "true"
        },
        {

```

```

        "targetPath":
"$ WorkplaceAddress.MobilePhoneNumberDescription",
        "sourcePath": "$ phoneNumbers[?(@.type ==
'mobile')].value",
        "optional": true
    },
    {
        "targetPath": "$ WorkplaceAddress.PhoneNumberDescription",
        "sourcePath": "$ phoneNumbers[?(@.type == 'work')].value",
        "optional": true
    },
    {
        "targetPath": "$ WorkplaceAddress.EmailURI",
        "sourcePath": "$ emails[0].value",
        "optional": true
    },
    {
        "constant": "SALES_REP",
        "targetPath": "$ Identity.BusinessRole[0].ID"
    },
    {
        "constant": "SALES_MANAGER",
        "targetPath": "$ Identity.BusinessRole[1].ID"
    }
]
}

```

### Using API version 3

By default, the Identity Provisioning uses the latest C4C API (SCIM based), for which property [c4c.api.version=3](#) by default.

Default transformation:

#### Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$ .id"
      },
      {
        "sourcePath": "$ .userName",
        "targetPath": "$ .userName"
      },
      {
        "sourcePath": "$ .emails[*].value",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$ .emails[?(@.value)]",
        "optional": true
      },
      {
        "constant": "work",
        "targetPath": "$ .emails[0].type"
      },
      {
        "sourcePath": "$ .userType",
        "targetPath": "$ .userType",
        "optional": true
      }
    ]
  }
}

```

```

        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName"
    },
    {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
    },
    {
        "sourcePath": $.active,
        "targetPath": $.active,
        "optional": true,
        "defaultValue": true
    },
    {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": $.schemas[0]
    },
    {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": $.schemas[1]
    },
    {
        "constant": "urn:ietf:params:scim:schemas:extension:sap:2.0:User",
        "targetPath": $.schemas[2]
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "optional": true
    }
]
},
// You cannot create groups but instead, you can create and update group
members, provided that groups in the source system have the exact same
display name
// as the groups in C4C. The group deletion operation is skipped.
"group": {
    "condition": "('%c4c.group.prefix%' == 'null') || ($.displayName =~ /
%c4c.group.prefix%.*/)",
    "skipOperations": [
        "delete"
    ],
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": $.id
        },
        {
            "sourcePath": $.displayName,
            "targetPath": $.displayName,
            "functions": [

```

```

        {
          "condition": "('%c4c.group.prefix%' != 'null') && (@ =~ /
%c4c.group.prefix%.*/)",
          "function": "replaceFirstString",
          "regex": "%c4c.group.prefix%",
          "replacement": ""
        }
      ]
    },
    {
      "sourcePath": "$.members[*].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.members[?(@.value)]",
      "functions": [
        {
          "function": "resolveEntityIds"
        }
      ],
      "defaultValue": []
    }
  ]
}

```

5. Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Blog: Employee Replication FAQ \(relevant for version 1\)](#) 

## 1.6.2.33 SAP SuccessFactors

Follow this procedure to set up SAP SuccessFactors as a target system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You have created a technical user with permissions to **call** the SAP SuccessFactors HCM Suite OData API to **import** employee data into an SAP SuccessFactors system. For more information, see [Permissions](#) and [URI Conventions \(OData Version 2.0\)](#) .
- You have the [Admin Center](#) > [Manage Permission Roles](#) > [Access to X.509 Certificates](#) permission (needed for configuring X.509 certificate-based authentication)

### Context

SAP SuccessFactors provides two APIs for its integration with Identity Provisioning: SAP SuccessFactors HCM Suite OData API and SAP SuccessFactors Workforce SCIM API. The value of `sf.api.version` property controls which API you use.

- When the value is set to **1**, or the property is not defined - SAP SuccessFactors HCM Suite OData API (in short, OData API) is used. This is the default value. SAP SuccessFactors source systems created before the introduction of `sf.api.version` property, use OData API.  
This version allows you to create and update **users**, as well as update **dynamic groups** and **group members**. To update a group from the source system, a group with the same name should already exist in SAP SuccessFactors.

#### ! Restriction

Note the following restrictions when using SAP SuccessFactors HCM Suite OData API:

- You cannot [create](#) or [delete](#) groups as these operations are currently not supported.
  - Managing SAP SuccessFactors **static groups** is not supported.
- When the value is set to **2** - SAP SuccessFactors Workforce SCIM API (in short, SCIM API) is used.  
This version allows you to provision static permission groups and user's group assignments. Provisioning of user's group assignments from a given source system to SAP SuccessFactors target system is possible if the user is an active employee with a work assignment and has a value of the `perPersonUuid` attribute. Alternatively, `personIdExternal` could be used as a fallback attribute.  
This requires that SAP SuccessFactors users are first provisioned with the attribute to the given system, for example Identity Authentication, where the `perPersonUuid` must be mapped to a custom attribute.

Afterwards, assigning users to groups in Identity Authentication and running a provisioning job would result in correct user assignments in SAP SuccessFactors target.

To update a group from the source system, a group with the same name should already exist in SAP SuccessFactors.

### ! Restriction

Note the following restrictions when using SAP SuccessFactors Workforce SCIM API:

- You cannot [create](#) or [delete](#) groups using the `Groups` APIs.
- The `Groups` patch API only supports updating membership of static permission groups.

For more information, see [Overview of SAP SuccessFactors Workforce System for Cross-Domain Identity Management API](#).

For more information about the difference between static and dynamic groups in SAP SuccessFactors, see [Permission Groups](#).

For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#).


You can configure [SAP SuccessFactors](#) as a target system to provision entities from a given source system. With regards to user termination, it is a standard HR process that is triggered from the SAP SuccessFactors system.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add [SAP SuccessFactors](#) as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Set up the communication between Identity Provisioning and SAP SuccessFactors and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP SuccessFactors target system, select the [Certificate](#) tab and choose [Generate](#) [Download](#) , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip this step if you use basic authentication. The next steps are performed in SAP SuccessFactors Admin Center and are relevant for certificate-based authentication only.

- b. Login to SAP SuccessFactors and go to [Admin Center](#). Follow the procedure described in [Upgrade to X.509 Certificate-Based Authentication for Incoming Calls](#).

Make sure you select [Identity Provisioning Service](#) in the [Integration Name](#) field.

4. Choose the [Properties](#) tab to configure the connection settings for your system.



## Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL to your SAP SuccessFactors API.</p> <p>For example:</p> <ul style="list-style-type: none"><li>For version 1: <a href="https://apitest.successfactors.com/odata/v2">https://apitest.successfactors.com/odata/v2</a></li><li>For version 2: <a href="https://apitest.successfactors.com">https://apitest.successfactors.com</a></li></ul> <p>To see the list of all SAP SuccessFactors data centers, see: <a href="#">HXM Suite OData APIs: API Endpoint URLs and System for Cross-domain Identity Management for Workforce in SuccessFactors</a> </p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"><li><a href="#">BasicAuthentication</a></li><li><a href="#">ClientCertificateAuthentication</a></li></ul>
(Optional) <code>sf.api.version</code>	<p>Handles the version of the API which is consumed by the SAP SuccessFactors system.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"><li><a href="#">1</a> - Indicates that SAP SuccessFactors HCM Suite OData API (in short, OData API) is used.</li><li><a href="#">2</a> - Indicates that SAP SuccessFactors Workforce SCIM API (in short, SCIM API) is used.</li></ul> <p>Default value: <a href="#">1</a></p>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">userID</a> of your SAP SuccessFactors technical user in the following format: <code>&lt;user_ID&gt;@&lt;company_ID&gt;</code></p>

Property Name	Description & Value
Password	<p>(Credential) Valid if <i>BasicAuthentication</i> is configured as authentication method.</p> <p>Enter the password for your SAP SuccessFactors technical user.</p>
<code>sf.company.id</code>	<p>Valid if <i>ClientCertificateAuthentication</i> is configured as authentication method.</p> <p>Enter the Company ID of your SAP SuccessFactors system.</p> <p>The Company ID is a short string of characters that identifies each SAP SuccessFactors system. It is like a username for your organization. All users of the same system share the same Company ID.</p>
<code>sf.user.attributes</code>	<p>Default property. It's a string representing a comma-separated list of user attributes by which the Identity Provisioning service retrieves the users that have to be updated in SAP SuccessFactors during the writing process. You can leave the default property value (all listed attributes), or leave only some of them.</p> <div> <p>→ Remember</p> <p>Always make sure that attribute <code>lastModifiedDateTime</code> is in the list. If you don't specify it, the provisioning to SAP SuccessFactors will fail.</p> </div> <p><b>Connector version:</b> SAP SuccessFactors version 1</p>

Property Name	Description & Value
<code>sf.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists on the target system. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s).</p> <p>This property defines by which unique attribute(s) the existing user to be searched (resolved). <b>If the service finds such a user on the target system via this filter, then the conflicting user will overwrite the existing one.</b> If the service does not find such a user, the creation will fail.</p> <p>According to your use case and system type, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property is missing during system creation. Its default value is <i>userName</i>. That means, if the service finds an existing user by a <i>userName</i>, it updates this user with the data of the conflicting one. If a user with such a <i>userName</i> is not found, the creation of the conflicting user fails.</li> <li>• Value = <i>emails[0].value</i>. If the service finds an existing user with such <i>email</i>, it updates this user with the data of the conflicting one. If a user with such <i>email</i> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> <li>• Value = <i>userName,emails[0].value</i>. If the service finds an existing user with both these <i>userName</i> and <i>email</i>, it updates this user with the data of the conflicting one. If such a user is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>userName</i></li> <li>• <i>emails[0].value</i></li> <li>• <i>userName,emails[0].value</i></li> <li>• <i>externalId</i>, or another SCIM attribute, or a conjunction of SCIM attributes</li> </ul> <p>Default value: <i>userName</i></p> <p><b>Connector version:</b> SAP SuccessFactors version 2</p>
(Optional) <code>sf.user.attributes.expand</code>	<p>This property writes additional user data related to <i>complex attributes</i>, which are specified in <code>sf.user.attributes</code>.</p> <p>Default value: <i>personKeyNav</i></p> <p><b>Connector version:</b> SAP SuccessFactors version 1</p>

Property Name	Description & Value
<code>sf.group.unique.attribute</code>	<p>If the service tries to create a group that already exists in the target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific).</p> <p>To make the search filter by a specific attribute, specify this attribute as a value for the <code>sf.group.unique.attribute</code> property.</p> <p>If the property is not specified, the search is done by the default attribute: <i>displayName</i></p> <p><b>Connector version:</b> SAP SuccessFactors version 2</p>
(Optional) <code>sf.group.prefix</code>	<p>This property distinguishes SAP SuccessFactors groups by specific prefix. It is an optional property which does not appear by default at system creation.</p> <p>Example value: <b>SF_</b></p> <p>You can use the example value or provide your own.</p> <p>When <b>set in the target system</b>, only groups containing the <b>SF_</b> prefix in their display name will be provisioned to SAP SuccessFactors. Groups without this prefix in the display name won't be provisioned.</p> <p>If the property is not set, the SAP SuccessFactors groups will be read and provisioned to the target system with their actual display names.</p> <p><b>Connector version:</b> SAP SuccessFactors version 2</p>

Property Name	Description & Value
(Optional) <code>sf.support.patch.operation</code>	<p>This property controls how modified users in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>• If set to <i>true</i>, PATCH operations are used to update users in the target system. This means, for example, that if a user attribute is modified, only this change will be provisioned and applied in the target system.</li> <li>• If set to <i>false</i>, PUT operations are used to update users in the target system. This means, for example, that if a user attribute is modified, all user attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p>Users can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>• In the source system, some user attributes are modified, or new attributes are added.</li> <li>• In the source system, a condition or a filter is set for users not to be read anymore.</li> <li>• A user is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <code>deleteEntity</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <i>true</i></li> <li>• <i>false</i></li> </ul> <p>Default value: <i>false</i></p> <p><b>Connector version:</b> SAP SuccessFactors version 2</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination (configuration):

Type=*HTTP*

Authentication=*BasicAuthentication*

ProxyType=*Internet*

URL=*https://apitest.successfactors.com/odata/v2*

User=*sfsf\_admin@mycompany.com*

Password=*\*\*\*\*\**

*sf.user.attributes=userId,username,addressLine1,jobTitle,lastName,country,email,location,firstName,lastModifiedDateTime,personKeyNav,manager/username*

*sf.user.attributes.expand=personKeyNav,manager*

---

## 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP SuccessFactors](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

When you configure SAP SuccessFactors as a target system, the Identity Provisioning service will read all user attributes from the selected source system and write them as user records in SAP SuccessFactors. Groups can be only updated if such already exist in SAP SuccessFactors.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP SuccessFactors. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP SuccessFactors HCM Suite OData API](#)

[SAP SuccessFactors Workforce SCIM API](#)

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

### Default transformation for SAP SuccessFactors HCM Suite OData API version 1:

#### Code Syntax

```
{
  "user": {
    "skipOperations": [
      "delete"
    ],
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.userId"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userId",
        "scope": "createEntity"
      }
    ]
  }
}
```

```

    },
    {
      "constant": "t",
      "targetPath": "$.status"
    },
    {
      "condition": "$.active == false",
      "constant": "f",
      "targetPath": "$.status"
    },
    {
      "sourcePath": $.userName,
      "optional": true,
      "targetPath": $.username
    },
    {
      "sourcePath": $.name.familyName,
      "optional": true,
      "targetPath": $.lastName
    },
    {
      "sourcePath": $.name.givenName,
      "optional": true,
      "targetPath": $.firstName
    },
    {
      "sourcePath": $.name.honorificPrefix,
      "optional": true,
      "targetPath": $.salutation
    },
    {
      "sourcePath": $.name.honorificSuffix,
      "optional": true,
      "targetPath": $.suffix
    },
    {
      "sourcePath": $.emails[0].value,
      "optional": true,
      "targetPath": $.email
    },
    {
      "condition": "$.emails[?(@.primary == true)].value != []",
      "sourcePath": "$.emails[?(@.primary == true)].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": $.email,
      "functions": [
        {
          "function": "elementAt",
          "index": 0
        }
      ]
    },
    {
      "sourcePath": $.timezone,
      "optional": true,
      "targetPath": $.timeZone
    },
    {
      "sourcePath": $.nickName,
      "optional": true,
      "targetPath": $.nickname
    },
    {
      "sourcePath": $.addresses[0].country,
      "optional": true,
      "targetPath": $.country
    },
  ],

```

```

    {
      "condition": "$.addresses[?(@.primary == true)].country !=",
      "sourcePath": "$.addresses[?(@.primary == true)].country",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.country",
      "functions": [
        {
          "function": "elementAt",
          "index": 0
        }
      ]
    },
    {
      "sourcePath": "$.addresses[0].locality",
      "optional": true,
      "targetPath": "$.city"
    },
    {
      "condition": "$.addresses[?(@.primary == true)].locality !=",
      "sourcePath": "$.addresses[?(@.primary == true)].locality",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.city",
      "functions": [
        {
          "function": "elementAt",
          "index": 0
        }
      ]
    },
    {
      "sourcePath": "$.addresses[0].formatted",
      "optional": true,
      "targetPath": "$.addressLine1"
    },
    {
      "sourcePath": "$.addresses[1].formatted",
      "optional": true,
      "targetPath": "$.addressLine2"
    },
    {
      "sourcePath": "$.addresses[2].formatted",
      "optional": true,
      "targetPath": "$.addressLine3"
    },
    {
      "condition": "$.phoneNumbers[?(@.type == 'work')].value !=",
      "sourcePath": "$.phoneNumbers[?(@.type == 'work')].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.businessPhone",
      "functions": [
        {
          "function": "elementAt",
          "index": 0
        }
      ]
    },
    {
      "condition": "$.phoneNumbers[?(@.type == 'fax')].value !=",
      "sourcePath": "$.phoneNumbers[?(@.type == 'fax')].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,

```



```

        "targetPath": "$.fax",
        "functions": [
            {
                "function": "elementAt",
                "index": 0
            }
        ]
    },
    {
        "condition": "$.phoneNumbers[?(@.type == 'mobile')].value !=
= []",
        "sourcePath": "$.phoneNumbers[?(@.type ==
'mobile')].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.cellPhone",
        "functions": [
            {
                "function": "elementAt",
                "index": 0
            }
        ]
    },
    {
        "condition": "$.phoneNumbers[?(@.type == 'home')].value !=
[]",
        "sourcePath": "$.phoneNumbers[?(@.type == 'home')].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.homePhone",
        "functions": [
            {
                "function": "elementAt",
                "index": 0
            }
        ]
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true,
        "targetPath": "$.empId"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true,
        "targetPath": "$.division"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true,
        "targetPath": "$.department"
    }
]
},
// By default, the group mapping is inactive (ignored) but groups are
supported.
// To start provisioning groups, either delete the statement "ignore":
true, or set its value to false.
"group": {
    "ignore": true,
    "skipOperations": [
        "delete"
    ]
}

```

```

    ],
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.groupID"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.groupName"
      },
      {
        "constant": "DynamicGroup",
        "targetPath": "$.__metadata.uri"
      },
      {
        "constant": "permission",
        "targetPath": "$.groupType"
      },
      {
        "sourcePath": "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]",
        "functions": [
          {
            "type": "resolveEntityIds"
          }
        ]
      }
    ]
  }
}

```

#### Default transformation for SCIM API version 2:

##### Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": [
          "urn:ietf:params:scim:schemas:extension:successfactors:2.0:User",
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:ietf:params:scim:schemas:extension:sap:2.0:User",
          "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.userType",
        "targetPath": "$.userType"
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",

```

```

        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid'],
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid'],
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails"
      },
      {
        "condition": "$.emails.length() > 0",
        "targetPath": "$.emails[*].type",
        "constant": "work"
      },
      {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.name.familyName"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.name.givenName"
      },
      {
        "sourcePath": "$.name.middleName",
        "optional": true,
        "targetPath": "$.name.middleName"
      },
      {
        "sourcePath": "$.name.honorificPrefix",
        "optional": true,
        "targetPath": "$.name.honorificPrefix"
      },
      {
        "sourcePath": "$.name.honorificSuffix",
        "optional": true,
        "targetPath": "$.name.honorificSuffix"
      },
      {
        "sourcePath": "$.name.formatted",
        "optional": true,
        "targetPath": "$.name.formatted"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['perPersonUuid'],
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['perPersonUuid']"
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['loginMethod']",

```

```

        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['loginMethod']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['personIdExternal']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['personIdExternal']"
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['customFields']",
        "optional": true,
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:successfactors:2.0:User']
['customFields']"
    },
    {
        "sourcePath": "$.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.phoneNumbers"
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active"
    },
    {
        "sourcePath": "$.groups[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.groups[?(@.value)]",
        "functions": [
            {
                "entityType": "group",
                "function": "resolveEntityIds"
            }
        ]
    }
]
},
"group": {
    "condition": "('%sf.group.prefix%' === 'null') || ($.displayName =~ /
%sf.group.prefix%.*/)",
    "skipOperations": [
        "delete"
    ],
    "mappings": [
        {
            "targetPath": "$.id",
            "sourceVariable": "entityIdTargetSystem"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName",
            "scope": "createEntity",
            "functions": [
                {
                    "condition": "('%sf.group.prefix%' !== 'null') && (@ =~ /
%sf.group.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%sf.group.prefix%",

```

```

        "replacement": ""
      }
    ],
    {
      "targetPath": "$.members",
      "type": "remove"
    },
    {
      "sourcePath": "$.members[*].value",
      "targetPath": "$.members[?(@.value)]",
      "optional": true,
      "preserveArrayWithSingleElement": true,
      "functions": [
        {
          "type": "resolveEntityIds"
        }
      ]
    }
  ]
}

```

6. Now, add a source system to read users from it. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[URI Conventions \(OData Version 2.0\)](#) ➔

[SAP SuccessFactors HCM Suite OData API](#)

[SAP SuccessFactors Workforce SCIM API](#)

## 1.6.2.34 SAP SuccessFactors Learning

Follow this procedure to set up SAP SuccessFactors Learning as a target system.

### Prerequisites

You have created a technical user with administrator permissions that will be used to call the SAP SuccessFactors Learning API for provisioning user information.

### Context

SAP SuccessFactors Learning is a learning solution which helps organizations to improve employee skills and talent management, align learning outcomes with performance goals, boost compliance, and train external audiences.

You can use the Identity Provisioning user interface (UI) to configure SAP SuccessFactors Learning as a target system where you can provision users from source systems.

#### ! Restriction

Provisioning **groups** is not supported.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SAP SuccessFactors Learning* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

#### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your SAP SuccessFactors Learning system. It follows the pattern: <code>https://&lt;root URL&gt;/learning/public-api/rest/admin/Integration.svc/ias</code>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the technical user ID for SAP SuccessFactors Learning.
Password	(Credential) Enter the password for the SAP SuccessFactors Learning technical user. For more information, see <a href="#">Learning Technical User</a> .
(Optional) <code>lms.user.unique.attribute</code>	This property appears by default when the system is created, and its value is set to <a href="#">userName</a> .  It defines by which unique attribute(s) an existing user to be resolved in the event of conflicting users.
(Optional) <code>lms.support.patch.operation</code>	Controls how modified users in the source system are updated in the target system. <ul style="list-style-type: none"> <li>If set to <a href="#">true</a>, PATCH operations are used to update users in the target system.</li> <li>If set to <a href="#">false</a>, PUT operations are used to update users in the target system.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 4. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP SuccessFactors Learning](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP SuccessFactors Learning system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: SAP SuccessFactors Learning](#) 

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target SAP SuccessFactors Learning entity.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": [
          "urn:ietf:params:scim:schemas:core:2.0:User", "urn:sap:cloud:scim:schemas:extension:custom:2.0:User", "urn:ietf:params:scim:schemas:extension:sap:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "optional": true
      },
      {
        "sourcePath": "$.externalId",
        "targetPath": "$.externalId",
        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]",
        "optional": true
      },
      {
        "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ][ 'siteID' ]",
        "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ][ 'siteID' ]",
        "optional": true
      },
      {
        "sourcePath": "$.custom-column-path-1",
        "targetPath": "$[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ]
[ 'customColumns' ][ '110' ]",
        "optional": true
      },
      {
        "sourcePath": "$.custom-column-path-2",
        "targetPath": "$[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ]
[ 'customColumns' ][ '120' ]",
        "optional": true
      }
    ]
  }
}
```



```

        "condition": "$.emails[?(@.primary== true)] empty false",
        "sourcePath": "$.emails[?(@.primary== true)].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.emails[?(@.value)]"
    },
    {
        "condition": "$.emails[?(@.primary== true)] empty true",
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.emails[0].value",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.name.givenName",
        "targetPath": "$.name.givenName",
        "optional": true
    },
    {
        "sourcePath": "$.name.middleName",
        "targetPath": "$.name.middleName",
        "optional": true
    },
    {
        "sourcePath": "$.name.familyName",
        "targetPath": "$.name.familyName"
    },
    {
        "condition": "$.phoneNumbers[?(@.primary== true)] empty false",
        "sourcePath": "$.phoneNumbers[?(@.primary== true)].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.phoneNumbers[?(@.value)]"
    },
    {
        "condition": "$.phoneNumbers[?(@.primary== true)] empty true",
        "sourcePath": "$.phoneNumbers[0].value",
        "targetPath": "$.phoneNumbers[0].value",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active"
    }
]
}

```

5. Add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.35 Sales Cloud – Analytics & AI

Follow this procedure to set up a target connector for Sales Cloud – Analytics & AI.

### Prerequisites

You have technical user credentials for an Sales Cloud – Analytics & AI (in short, [SCAAI](#)) system with read and write access permissions.

### Context

After fulfilling the prerequisites, follow the procedure to add a target system for Sales Cloud – Analytics & AI to write users and user assignments to groups. This target system consumes SCIM 2.0 API provided by SCAAI.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *Sales Cloud – Analytics & AI* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

#### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the SCIM API portal of your SCAAI system.

Property Name	Value
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the user for your SCAAI system.
Password	Enter the password for your SCAAI user.
OAuth2TokenServiceURL	Enter the URL to the OAuth2 token service.  If not sure about the exact URL, ask your SCAAI administrator.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

Exemplary destination:

Type=[HTTP](#)

Authentication=[BasicAuthentication](#)

ProxyType=[Internet](#)

URL=[http://myscaai:8080/scim\\_services](#)

User=[MySCAAIUser](#)

Password=\*\*\*\*\*

OAuth2TokenServiceURL=[http://myscaai:8080/gateway\\_services/api/auth/ips/token](#)

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [Sales Cloud – Analytics & AI](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your Sales Cloud – Analytics & AI. For more information, see: [Manage Transformations \[page 1494\]](#)

**Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target entity. The Identity Provisioning takes a user's original [userName](#) (from the source system) and assigns it as a user ID in the target SCAAI system. Analogically, the Identity Provisioning takes a group's original [displayName](#) and assigns it as a group ID in SCAAI.

**Default transformation:**

≡ Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath": "$.userName",
      "targetPath": "$.externalId"
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.userName",
      "targetPath": "$.userName"
    }
  ]
},
"group": {
  "skipOperations": [
    "delete"
  ],
  "mappings": [
    {
      "sourceVariable": "entityIdTargetSystem",
      "targetPath": "$.id"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.externalId"
    },
    {
      "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath": "$.schemas[0]"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName"
    },
    {
      "sourcePath": "$.members[*].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath": "$.members[?(@.value)]",
      "functions": [
        {
          "function": "resolveEntityIds"
        }
      ]
    }
  ]
}
}

```

If you want the users and groups in SCAA to have the **same** IDs as the respective users and groups in the source system, modify the transformation mappings as follows:

- In the source system:

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "correlationAttribute": true
      },
      ...
    ]
  },
  "group": {

```

```

    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id"
      },
      ...
    ]
  }

```

- In the SCAAI target system:

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.externalId"
      },
      ...
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.externalId"
      },
      ...
    ]
  },
  ...
}

```

5. Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.36 Cloud Foundry UAA Server

Follow this procedure to set up a Cloud Foundry UAA Server as a target system.

### Prerequisites

#### ! Restriction

This system is available for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You have technical user credentials for a Cloud Foundry system with write access permissions. In case OAuth is used for authentication, client ID and secret are required when creating a destination for access token retrieval. You need Cloud Foundry UAA version **4.2** or higher
- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need to do this only if the Cloud Foundry UAA server is exposed in a private corporate network. For more information, see [Cloud Connector](#).

## Context

User Account and Authentication Service (UAA) is an OAuth2 server that you can use for centralized identity management. It owns the user accounts and authentication sources and supports standard protocols (such as [SAML](#), [LDAP](#), and [OpenID Connect](#)) to provide SSO and delegated authorization to Web applications. For more information, see [Cloud Foundry: Overview](#) .

### → Tip

This connector is meant for writing users and groups in **general** Cloud Foundry systems (they could be non-SAP ones). If you want to trigger provisioning of entities to SAP Business Technology Platform Cloud Foundry applications, you'd better use [SAP BTP XS Advanced UAA \(Cloud Foundry\) \[page 777\]](#) target system.

These target systems consume SCIM 1.1 API provided by Cloud Foundry UAA.

### → Remember

You can write users and groups to Cloud Foundry on **application** level only. You cannot provision or manage them on a [subaccount](#) level.

Follow the steps below to create Cloud Foundry UAA as a target system to provision users and groups.

## Procedure

1. (Optional) If the Cloud Foundry UAA server is exposed in a private corporate network, add an access control system mapping in Cloud Connector. For more information, see [Configure Access Control \(HTTP\)](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add [Cloud Foundry UAA Server](#) as a target system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the Cloud Foundry UAA SCIM API. If not sure about the exact URL, ask your Cloud Foundry UAA administrator.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
OAuth2TokenServiceURL	As you need to make OAuth authentication to the UAA system, enter the URL to the OAuth2 token service. If not sure about the exact URL, ask your Cloud Foundry UAA administrator.
User	Enter the OAuth client ID of the Cloud Foundry UAA technical user.
Password	(Credential) Enter the OAuth client secret of the technical user.
uaa.origin	Enter the location of your Cloud Foundry identity provider. If not sure about the value, ask your Cloud Foundry UAA administrator.  The value of this property is a string, which will be used as the <a href="#">origin</a> attribute in the system transformation.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

Exemplary destination:

Type=*HTTP*

Authentication=*BasicAuthentication*

ProxyType=*Internet*

URL=*https://api.authentication.hana.ondemand.com*

OAuth2TokenServiceURL=*https://MyCFaccount.authentication.hana.ondemand.com/oauth/token*

User=*MyCFuser*

Password=*\*\*\*\*\**

uaa.origin=*my\_UAA\_location*

---

## 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Cloud Foundry UAA Server* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

- **Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal Cloud Foundry UAA representation to the target entity.
- **User offboarding** – If a user has been deleted from the source system, this change is recognized, and the user is deleted from the Cloud Foundry UAA target system too.

You can change the default transformation mapping rules to reflect your current setup of entities in your Cloud Foundry UAA server. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Cloud Foundry UAA API: Users](#) ➡

[Cloud Foundry UAA API: Groups](#) ➡

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

### Default transformation:

#### Code Syntax

```
{
  "user": {
    "condition": "$.emails.length() > 0",
    "mappings": [
      {
        "constant": "uaa-dummy-value",
        "targetPath": "$.id",
        "scope": "createEntity"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      }
    ]
  }
}
```



```

        "sourcePath": "$.userName",
        "targetPath": "$.userName"
    },
    {
        "sourcePath": $.name",
        "targetPath": $.name",
        "optional": true
    },
    {
        "sourcePath": $.active",
        "targetPath": $.active",
        "optional": true
    },
    {
        "sourcePath": $.verified",
        "targetPath": $.verified",
        "optional": true
    },
    {
        "constant": "%uaa.origin%",
        "targetPath": $.origin"
    },
    {
        "sourcePath": $.phoneNumbers",
        "targetPath": $.phoneNumbers",
        "preserveArrayWithSingleElement": true,
        "optional": true
    },
    {
        "sourcePath": $.emails",
        "targetPath": $.emails",
        "preserveArrayWithSingleElement": true
    },
    {
        "condition": $.emails[?(@.primary == true)].value == [],
        "targetPath": $.emails[0].primary",
        "constant": true
    },
    {
        "constant": "urn:scim:schemas:core:1.0",
        "targetPath": $.schemas[0]"
    }
]
},
"group": {
    "condition": "('%uaa.group.prefix%' === 'null') || ($.displayName =~ /%uaa.group.prefix%.*/)",
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": $.id"
        },
        {
            "sourcePath": $.displayName",
            "targetPath": $.displayName",
            "functions": [
                {
                    "condition": "('%uaa.group.prefix%' !== 'null') && (@ =~ /%uaa.group.prefix%.*/)",
                    "function": "replaceFirstString",
                    "regex": "%uaa.group.prefix%",
                    "replacement": ""
                }
            ]
        }
    ]
},
{
    "sourcePath": $.description",
    "targetPath": $.description",

```

```

        "optional": true
      },
      {
        "sourcePath": "$.members",
        "targetPath": "$.members",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "functions": [
          {
            "function": "resolveEntityIds"
          },
          {
            "condition": "@.type EMPTY false",
            "function": "toUpperCaseString",
            "applyOnElements": true,
            "applyOnAttribute": "type",
            "locale": "en_EN"
          }
        ]
      },
      {
        "constant": "urn:scim:schemas:core:1.0",
        "targetPath": "$.schemas[0]"
      }
    ]
  }
}

```

6. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Cloud Foundry UAA: Users](#) ➡

[Cloud Foundry UAA: Groups](#) ➡

## 1.6.2.37 Google G Suite

Follow this procedure to set up Google G Suite as a target system.

### Prerequisites

#### ! Restriction

This system is available for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it only through **SAP Identity Access Governance** bundle option.

1. Sign in to the Google API console (<https://console.developers.google.com>) and create a project.
2. Enable the Admin SDK. To do this, go to ► [Dashboard](#) ► [ENABLE API](#) ► [Admin SDK](#) ► [ENABLE](#) ►.
3. Create a service account for your project. We recommend that you select [Enable G Suite Domain-wide Delegation](#) during the creation. If you skip this option, you can set it later. For more information, see [Creating a service account](#).
4. Then, in the Google admin console (<https://admin.google.com>), a user with **Super Admin** role can delegate domain-wide authority to your service account. This way, it will have access to the Google Admin SDK on behalf of your user. For more information, see [Delegating domain-wide authority](#).

#### i Note

When specifying the scopes, the administrator has to enter the following:

```
https://www.googleapis.com/auth/admin.directory.user, https://www.googleapis.com/auth/admin.directory.group
```

### Context

A Google service account with delegated domain-wide authority is required for authentication and authorization of the Identity Provisioning service to G Suite domain. The authentication is based on OAuth 2.0 protocol with JSON Web Token (JWT). The private key for the signature is distributed by Google via one-time downloadable JSON data, which is accessible by the domain administrator. The private key is encoded in PKCS8 format and is in the [private\\_key](#) field of the JSON data. For more information, see [JSON Web Token \(JWT\)](#).

- When using it as a source system, you can read both users and groups from Google G Suite and provision them to any target system you have added in the Identity Provisioning user interface.
- When using it as a target system, you can write both users and groups, read from any source system you have added in the Identity Provisioning user interface. Google G Suite can automatically create accounts for your users in the Google Cloud Datastore.

The Identity Provisioning service supports user and group operations based on the following Google Directory API. See the table below.

User Operations	Group Operations
<a href="#">Create a user</a> ➡	<a href="#">Create a group</a> ➡
<a href="#">Retrieve a user</a> ➡	<a href="#">Retrieve a group's properties</a> ➡
<a href="#">Update a user</a> ➡	<a href="#">Update a group's properties</a> ➡
<a href="#">Delete a user</a> ➡	<a href="#">Delete a group</a> ➡

### ⚠ Caution

You can only provision users whose e-mails are from verified domains.

If you have successfully finished with the initial setup (described in the **Prerequisites** section), continue with the procedure below.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *Google G Suite* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Specify the service URL:  <i><a href="https://www.googleapis.com/admin/directory">https://www.googleapis.com/admin/directory</a></i>

Property Name	Description & Value
ProxyType	Enter: <b>Internet</b>
Authentication	<p>Enter: <i>BasicAuthentication</i></p> <p>The authentication type in use is actually <b>OAuth</b> with JWT. But for any provisioning system based on OAuth, <b>BasicAuthentication</b> is used along with the OAuth2TokenServiceURL additional property.</p>
User	Enter the service account's ID. You can take it from the " <i>client_email</i> " field in the JSON data, downloaded during the setup of Google service account.
Password	Enter the service account's private key, which represents a long string in PKCS8 format. You can take it from the " <i>private key</i> " field in the JSON data, downloaded during the setup of Google service account.
OAuth2TokenServiceURL	To make OAuth authentication to the Google G Suite system, enter the URL to the access token provider service. For more information, see Using <a href="#">OAuth 2.0 to Access Google APIs</a> .
jwt.subject	<p>Enter the Google G Suite user on behalf of which the Google Directory API is called. This user has been assigned the role <b>User Management Admin</b>.</p> <p>This property corresponds to "sub" claim in JWT being generated during access token request: <a href="#">JWT: "sub" (Subject) Claim</a> .</p>
(Optional) jwt.scope	<p>Enter space-separated Google Directory API authorization scopes. For example:</p> <p><b><a href="https://www.googleapis.com/auth/admin.directory.user">https://www.googleapis.com/auth/admin.directory.user</a></b></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary Configuration:

```
ProxyType=Internet

Type=HTTP

Authentication=BasicAuthentication

URL=https://www.googleapis.com/admin/directory

User=1234567890-compute@developer.gserviceaccount.com

Password=-----BEGIN PRIVATE KEY-----\n123ABCDEFG123456789...
.../123456789ABCDEFG123=\n-----END PRIVATE KEY-----\n

OAuth2TokenServiceURL=https://www.googleapis.com/oauth2/v4/token

jwt.subject=john.smith@me123.accounts.ondemand.com

jwt.scope=https://www.googleapis.com/auth/admin.directory.user
```

---

#### 4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [Google G Suite](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

Transformation principles for the target system integration:

- **Mapping logic** – The provisioning framework reads all attributes from the intermediate JSON data and tries to create consistent records in the Google G Suite target system, using all the available attributes accepted by the Google Directory API. When a required attribute is missing, the default transformation is designed with a condition that will exclude the inconsistent records. Bear in mind the following:
  - Make sure that the JSON data sent by the source system is consistent with the configuration template of the target. For example, if the source system contains mandatory fields and the target one does not support such kind of data, then the target system skips these fields. This may cause crucial data loss.
  - There is a special user status type called **suspended** (temporarily blocks a user without deleting any account data) for the Google directory accounts. When the status of the user account is changed to **suspended**, the Google Directory API will not accept any changes on the user attributes. Once the suspended user is restored by the administrator, all attribute changes pending for the account will be successfully provisioned with the next provisioning job.

#### Caution

An initial password setup is mandatory for all newly provisioned users. This is required by the Google G Suite API and must be provided when new accounts are created. The constant value that you see as configuration for the password attribute in the default transformation is generated by SAP. You have to change the constant value with another one, known only by the representatives of your company, before starting to use the Identity Provisioning service for creating users in your corporate Google G Suite system automatically.

- **User off-boarding** – Identity Provisioning service is handling the deletion status of the users. When a user is deleted from the source system, this deletion will be enforced into the Google G Suite system as well.

You can change the default transformation mapping rules to reflect your current setup of entities in your Google G Suite. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Google Directory API: Users](#) ➦

[Google Directory API: Groups](#) ➦

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "condition": "($.emails.length() > 0) && ($.name.familyName EMPTY
false)",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name"
      },
      {
        "sourcePath": "$.emails[0].value",
        "targetPath": "$.primaryEmail"
      },
      {
        "sourcePath": "$.phoneNumbers",
        "optional": true,
        "targetPath": "$.phones"
      },
      {
        "targetPath": "$.password",
        "scope": "createEntity",
        "functions": [
          {
            "type": "randomPassword",
            "passwordLength": 16,
            "minimumNumberOfLowercaseLetters": 1,
            "minimumNumberOfUppercaseLetters": 1,
            "minimumNumberOfDigits": 1,
            "minimumNumberOfSpecialSymbols": 0
          }
        ]
      },
      {
        "constant": "false",
        "targetPath": "$.suspended"
      },
      {
        "condition": "$.active == false",
        "constant": true,
        "targetPath": "$.suspended"
      },
      {
        "constant": "true",
        "targetPath": "$.changePasswordAtNextLogin"
      }
    ]
  },
  "group": {
    "ignore": true,
```

```

    "mappings": [
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath": "$.schemas[0]"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.name"
      }
    ],
    // Google G Suite requires a group e-mail. By default, the email attribute
    // is mapped to displayName. If group's Display Name does not contain an e-
    // mail,
    // you can either map email to another attribute, or concatenate
    // displayName with your domain. To learn how, see the detailed explanation
    // and example below.
    {
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.email",
        "scope": "createEntity"
      },
      {
        "sourcePath": "$.members[?(@.type == 'User')].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.id)]",
        "functions": [
          {
            "entityType": "user",
            "type": "resolveEntityIds"
          }
        ]
      }
    ]
  }
}

```

If the **displayName** attribute in the source system transformation does not provide group e-mails, you can modify the transformation the following ways:

- Map **email** to another attribute that contains a unique group e-mail.
- Concatenate the **displayName** attribute with your domain. For example:

#### Sample Code

```

{
  "sourcePath": "$.displayName",
  "targetPath": "$.email",
  "scope": "createEntity",
  "functions": [
    {
      "type": "concatString",
      "suffix": "@test.myaccount.ondemand.com"
    }
  ]
}

```



5. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.38 LDAP Server

Follow this procedure to set up LDAP Server as a target system.

## Prerequisites

### ! Restriction

This system is available for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it only through **SAP Identity Access Governance** bundle option.

### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#) or [Cloud Connector \(Cloud Foundry\)](#)
- **For tenants running on the infrastructure of SAP Cloud Identity Services:** You have a multi-environment subaccount in the Cloud Foundry region that maps the region of your Identity Authentication tenant and it is subscribed to the [Cloud Identity Services](#) application. For more information, see [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure](#).
- You have the credentials of a technical user in the LDAP Server, which is used to call the LDAP Server API to write users and their attributes.

## Context

You can use LDAP Server to write entities retrieved from a source system. This scenario supports writing **users** and **group assignments**.

There are two versions of the LDAP Server connector. Both consume the LDAP Server API to read and write users and groups. The versions are handled by the `ldap.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on May 25, 2023) LDAP Server API version 1 is used. This is the default value of `ldap.api.version`.  
When using this version of the connector, the entities (users and groups) are read with all attributes.
- When the value is set to **2** – LDAP Server API version 2 is used.  
This version of the connector comes with improved performance of the read operation for user and group attributes. You are now able to define which user and group attributes to be read. This is possible by adding values to the properties `ldap.user.attributes` or `ldap.group.attributes`.  
Via these properties, you are able to add also user and group operational attributes (attributes which the directory organizes for internal use). For more information, refer to the official LDAP server documentation. After the additional values of the properties are set, the default read or proxy read transformations should also be adjusted accordingly.  
For more information on how to update to LDAP Server connector version 2, see [Update Connector Version \[page 1484\]](#).

## Procedure

1. Add an access control system mapping for the **LDAP Server** in the Cloud Connector. This is needed to allow the Identity Provisioning service to access the LDAP server as a back-end system on the intranet. For more information, see [Configure Access Control \(LDAP\)](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add **LDAP Server** as a target system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the **Properties** tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the **Destination Name** combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the **Properties** tab, the value set in the **Properties** tab is considered with higher priority.

We recommend that you use the **Properties** tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">LDAP</a>
ldap.url	Specify the destination URL. It must be in the following format:  ldap://<external_host>:<external_port>
ldap.proxyType	Enter: <a href="#">OnPremise</a>
ldap.authentication	Enter: <a href="#">BasicAuthentication</a>
ldap.user	Enter the <a href="#">distinguishedName</a> of the technical LDAP user. This is the user you need to establish the connection and to perform all queries.
ldap.password	(Credential) Enter the password for the LDAP technical user.
ldap.group.path	Enter the complete path to the node containing the groups in the LDAP tree.
ldap.user.path	Enter the complete path to the users in the LDAP tree.
(Optional)ldap.api.version	Defines the version of LDAP Server API.  <b>Possible values:</b> <ul style="list-style-type: none"> <li>• <b>1</b> - Indicates that LDAP Server API version 1 is used.</li> <li>• <b>2</b> - Indicates that LDAP Server API version 2 is used.</li> </ul> If the property is not defined - LDAP Server API version 1 is used.
CloudConnectorLocationId	Relevant when the proxy type is <a href="#">OnPremise</a> . Use it only if your SAP Business Technology Platform account uses more than one Cloud Connector.

### → Remember

We strongly recommend that you enter different paths for LDAP users and groups. That means, the value of `ldap.user.path` should be different than the value of `ldap.group.path`.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

The LDAP Server target system is created by default with the properties listed below:

#### Default LDAP Properties

```
ldap.user.attributes=  
ldap.user.object.class= inetOrgPerson  
ldap.group.object.class= groupOfNames  
ldap.group.uniquename.attribute= cn  
ldap.attribute.group.id=cn  
ldap.attribute.group.member= member  
ldap.attribute.group.object.class.required=cn  
ldap.attribute.user.object.class.required=cn  
ldap.attribute.user.id= uid  
ldap.attribute.dn=distinguishedName  
ldap.page.size= 100  
ldap.attribute.user.mail= mail  
ldap.attribute.user.mobile=mobile  
ldap.attribute.user.givenName= givenName  
ldap.attribute.user.surname= sn  
ldap.attribute.user.groups= memberOf  
ldap.attribute.user.telephoneNumber= telephoneNumber
```

---

#### **i** Note

The **ldap.attribute.\*** properties are used as parameterized properties in the default transformation. That is, if a property used in the transformation doesn't have a value, the provisioning job will fail when the transformation is loaded on runtime and the property value is substituted.

Also, you can change a property and use a new one (with a new name). In this case, you must replace the old property with the new one at all corresponding places in the transformation.

#### **i** Note

The **ldap.attribute.dn** property is automatically assigned and cannot be configured. It is used in the target system configuration for conflict resolution during user or group provisioning by the service.

#### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [LDAP Server](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your LDAP server. For more information, see [Manage Transformations \[page 1494\]](#).

Currently, the default write transformations of the two connector versions have no differences, so there is no need to update them.

Before the write transformation (in the intermediate JSON data), the entity attributes are in SCIM format. After the transformation, the attributes in the LDAP Server are represented as arrays (single-element arrays, or multi-value arrays separated by comma (,)). For more information, see the official documentation for LDAP Server schema attributes in the **Related Information** section.

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.%ldap.attribute.user.id%[0]",
        "targetVariable": "entityIdTargetSystem",
        "scope": "createEntity"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.%ldap.attribute.user.object.class.required%[0]"
      },
      {
        "sourcePath": "$.emails[*].value",
        "optional": true,
        "targetPath": "$.%ldap.attribute.user.mail%"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.%ldap.attribute.user.givenName%[0]"
      },
      {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.%ldap.attribute.user.surname%[0]"
      },
      {
        "sourcePath": "$.phoneNumbers[*].value",
        "optional": true,
        "targetPath": "$.%ldap.attribute.user.mobile%"
      }
    ]
  },
  "group": {
    "ignore": true,
    "mappings": [
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.%ldap.attribute.group.id%[0]",
        "targetVariable": "entityIdTargetSystem",
        "scope": "createEntity"
      },
      {
        "sourcePath": "$.displayName",
        "targetPath": "$.%ldap.attribute.group.object.class.required%[0]"
      }
    ]
  }
}
```

```

    },
    {
      "constant": [],
      "targetPath": "$.member"
    },
    {
      "sourcePath": "$.members[*]",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetVariable": "membersVariable",
      "functions": [
        {
          "condition": "@.type != 'Group'",
          "entityType": "user",
          "type": "resolveEntityIds"
        },
        {
          "condition": "@.type == 'Group'",
          "entityType": "group",
          "type": "resolveEntityIds"
        },
        {
          "condition": "@.type != 'Group'",
          "function": "concatString",
          "applyOnElements": true,
          "applyOnAttribute": "value",
          "prefix": "%ldap.attribute.user.id%",
          "suffix": ",%ldap.user.path%"
        },
        {
          "condition": "@.type == 'Group'",
          "function": "concatString",
          "applyOnElements": true,
          "applyOnAttribute": "value",
          "prefix": "%ldap.attribute.group.id%",
          "suffix": ",%ldap.group.path%"
        }
      ]
    }
  ],
  {
    "sourceVariable": "membersVariable",
    "preserveArrayWithSingleElement": true,
    "optional": true,
    "targetPath": "$.member",
    "variablePath": "$[*].value"
  }
]
}

```

Below is illustrated an example of how the data from LDAP Server looks like before and after executing a certain mapping from the write transformation:

Transformation Snippet (from the group mapping)	Intermediate JSON Data (before the transformation)	Target JSON Data (written in the LDAP Server)
<div> <div>Sample Code</div> <pre> "group": {   ...   {     "sourcePath":     "\$.members[*]",     "preserveArrayWithSingleElement": true,     "optional": true,     "targetVariable":     "membersVariable",     "functions": [       {         "condition":         "@.type != 'Group'",         "entityType":         "user",         "type":         "resolveEntityIds"       },       {         "condition":         "@.type == 'Group'",         "entityType":         "group",         "type":         "resolveEntityIds"       },       {         "condition":         "@.type != 'Group'",         "function":         "concatString",         "applyOnElements": true,         "applyOnAttribute":         "value",         "prefix":         "%ldap.attribute.user.id%",         "suffix":         ",%ldap.user.path%"       }     ]   } } </pre> </div>	<div> <div>Sample Code</div> <pre> ... "members": [   {     "value":     "SALES_US"   },   {     "value":     "SALES_EU"   },   {     "value":     "SALES_JA"   } ] ... </pre> </div>	<div> <div>Sample Code</div> <pre> ... "member": [   "SALES_US",   "SALES_EU",   "SALES_JA" ] ... </pre> </div>

Transformation Snippet (from the group mapping)	Intermediate JSON Data (before the transformation)	Target JSON Data (written in the LDAP Server)
<pre> "condition": "@.type == 'Group'",   "function":     "concatString",  "applyOnElements": true,  "applyOnAttribute": "value",   "prefix": "%ldap.attribute.group.id%",   "suffix": ",%ldap.group.path%"     }   },   {  "sourceVariable": "membersVariable",  "preserveArrayWithSingleElement": true,   "optional": true,   "targetPath": "\$member",   "variablePath": "\$[*].value"     }   ] } </pre>		

### Note

By default, the **cn** attribute is used for writing the groups. An administrator can change this behavior by setting the following properties:

- `ldap.group.uniquename.attribute` – the value can be either the CN or the whole DN (**distinguishedName**) of the group.
- `ldap.attribute.group.id` – the value can be CN or another attribute to be used as a group ID instead (for example, **displayName** or **description**).

For more information about these properties, see: [List of Properties \[page 94\]](#)

- Now, add a source system to read users and groups from it. Choose from: [Source Systems \[page 452\]](#)

## Related Information

[Technical Documents](#) ➔



## 1.6.2.39 Microsoft Active Directory

Follow this procedure to set up Microsoft Active Directory as a target system.

### Prerequisites

#### ! Restriction

This system is available for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it only through **SAP Identity Access Governance** bundle option.

#### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)
- You have the credentials of a technical user in the Microsoft Active Directory, which is used to call the Microsoft Active Directory API to write users, groups and their attributes.

### Context

You can configure Microsoft Active Directory as a target system to provision identities. Currently, you can provision users, groups and group assignments.

There are two versions of the Microsoft AD connector. Both consume the LDAP Server API to read and write users and groups. The versions are handled by the `ldap.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on June 19, 2023) LDAP Server API version 1 is used. This is the default value of `ldap.api.version`.  
When using this version of the connector, the entities (users and groups) are read with all attributes. In this version, the `group members` attribute mapping in the proxy read transformation does not include `type` sub-attribute. In this case, all members are considered of type `user`, which is the sub-attribute fallback value. As a consequence, if the external system includes nested groups, they will not be handled properly.
- When the value is set to **2** – LDAP Server API version 2 is used.

This version of the connector is with improved performance of the read operation for user and group attributes. You are now able to define which user and group attributes to be read. This is possible by adding values to the properties `ldap.user.attributes` or `ldap.group.attributes`.

Via these properties, you are able to add also user and group operational attributes (attributes which the directory organizes for internal use). For more information, refer to the official LDAP server documentation. After the additional values of the properties are set, the default read or proxy read transformations should also be adjusted accordingly.

In this version, the `group members` attribute mapping in the proxy read transformation is enhanced with `type` sub-attribute. The sub-attribute has two possible values – `User` and `Group`. This allows you to read and preserve nested groups.

For more information on how to update to Microsoft Active Directory version 2, see [Update Connector Version \[page 1484\]](#).

## Procedure

1. Add an access control system mapping for the **Microsoft Active Directory** in the Cloud Connector. This is needed to allow the Identity Provisioning service to access Microsoft AD as a back-end system on the intranet. For more information, see [Configure Access Control \(LDAP\)](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add [Microsoft Active Directory](#) as a target system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">LDAP</a>
<code>ldap.url</code>	Specify a destination URL. It must be in the following format:  <code>ldap://&lt;ext_host&gt;:&lt;ext_port&gt;</code>
<code>ldap.proxyType</code>	Enter: <a href="#">OnPremise</a>

Property Name	Description & Value
<code>ldap.authentication</code>	Enter: <a href="#">BasicAuthentication</a>
<code>ldap.user</code>	Enter the <a href="#">distinguishedName</a> or the <a href="#">userPrincipalName</a> of the Microsoft AD technical user. This is the user you need to establish the connection and to perform all queries.
<code>ldap.password</code>	(Credential) Enter the password for the Microsoft AD technical user.
<code>ldap.attribute.user.id</code>	Default property, which denotes the ID of a user. By default, it's set to: <a href="#">cn</a>
<code>ldap.attribute.group.id</code>	Default property, which denotes the ID of a group. By default, it's set to: <a href="#">cn</a>
<code>ldap.attribute.dn</code>	Default property, which denotes the distinguished name of a user or a group.  The distinguished name is automatically assigned and cannot be configured. It is used in the target system configuration for conflict resolution during user or group provisioning by the service.  Only possible value: <a href="#">distinguishedName</a>
<code>ldap.group.path</code>	Enter the complete path to the node containing the groups in Microsoft Active Directory.
<code>ldap.user.path</code>	Enter the complete path to the users in Microsoft Active Directory.
<code>CloudConnectorLocationId</code>	Relevant when the proxy type is <a href="#">OnPremise</a> . Use it only if your SAP Business Technology Platform account uses more than one Cloud Connector.

Example for a destination or a set of properties:

```
Type=LDAP

Name=MyADDestination

ldap.user=john.smith@some.dummy.domain.com

ldap.password=*****

ldap.attribute.user.id=cn

ldap.attribute.group.id=cn

ldap.attribute.dn=distinguishedName

ldap.url=ldap://abcd:123

ldap.proxyType=OnPremise

ldap.authentication=BasicAuthentication

ldap.group.path=OU=Groups,OU=IAS,DC=global,DC=corp,DC=mycompany

ldap.user.path=OU=Users,OU=IAS,DC=global,DC=corp,DC=mycompany
```

---

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

#### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Microsoft Active Directory* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

In the intermediate JSON data, the Microsoft Active Directory attributes are in SCIM format. After the write transformation, the attributes are represented as arrays – single-element arrays, or multi-value arrays separated by comma (.). For more information about Microsoft AD schema attributes, see the **Related Information** section.

Currently, the default write transformations of the two connector versions have no differences, so there is no need to update them.

You can change the default transformation mapping rules to reflect your current setup of entities in Microsoft AD. For more information, see:

[Manage Transformations \[page 1494\]](#)

[MS Graph: Users](#) ➡

[MS Graph: Groups](#) ➡

#### **i** Note

You can provision users with *unicodePwd* attribute to Microsoft Active Directory target systems although it is not included in the default write transformation. This attribute specifies the password of the user in Windows NT operating system one-way format (OWF).

To provision the user password in encrypted format, proceed as follows:

1. Add the unicodePwd attribute mapping in the write transformation. For example:

#### Code Syntax

```
{
  "sourcePath": "$.Source_System_Attribute",
  "targetPath": "$.unicodePwd"
}
```

2. Follow the requirements for provisioning users with [unicodePwd](#), as described in [unicodePwd](#) 

#### Default transformation:

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "condition": "('%ldap.attribute.user.id%' != '%ldap.attribute.dn%')",
        "sourcePath": "$.userName",
        "targetPath": "$.ldap.attribute.user.id[0]",
        "targetVariable": "entityIdTargetSystem",
        "scope": "createEntity"
      },
      // If a user is not a direct member of the configured user base
      // path, then its distinguishedName is configured to be equal to CN =
      // <userName>,<nested_path>,<base_path>,
      // where <nested_path> is read from "sourcePath": "$
      [ 'urn:sap:cloud:scim:schemas:extension:ad:2.0:User' ][ 'nestedPath' ]"
      {
        "condition": "('%ldap.attribute.user.id%' == '%ldap.attribute.dn%')",
        "sourcePath": "$
      [ 'urn:sap:cloud:scim:schemas:extension:ad:2.0:User' ][ 'nestedPath' ]",
        "optional": true,
        "targetVariable": "nestedPathVariable",
        "defaultValue": "",
        "scope": "createEntity"
      },
      {
        "condition": "('%ldap.attribute.user.id%' == '%ldap.attribute.dn%')",
        "sourcePath": "$.userName",
        "functions": [
          {
            "function": "concatString",
            "prefix": "CN="
          },
          {
            "condition": "('${nestedPathVariable}' != '')",
            "function": "concatString",
            "suffix": ","
          },
          {
            "condition": "('${nestedPathVariable}' != '')",
            "function": "concatString",
            "suffix": "${nestedPathVariable}"
          }
        ],
        "function": "concatString",
```

```

        "suffix": ",%ldap.user.path%"
    }
],
"targetPath": "$.%ldap.attribute.user.id%[0]",
"targetVariable": "entityIdTargetSystem",
"scope": "createEntity"
},
{
    "sourcePath": "$.userName",
    "targetPath": "$.cn[0]",
    "scope": "createEntity"
},
{
    "sourcePath": "$.userName",
    "targetPath": "$.sAMAccountName[0]"
},
{
    "sourcePath": "$.displayName",
    "targetPath": "$.displayName[0]",
    "defaultValue": [],
    "optional": true
},
{
    "sourcePath": "$.emails[0].value",
    "targetPath": "$.mail[0]",
    "defaultValue": [],
    "optional": true
},
{
    "sourcePath": "$.name.givenName",
    "targetPath": "$.givenName[0]",
    "defaultValue": [],
    "optional": true
},
{
    "sourcePath": "$.name.familyName",
    "targetPath": "$.sn[0]",
    "defaultValue": [],
    "optional": true
}
]
},
"group": {
    "ignore": true,
    "mappings": [
        {
            "condition": "('%ldap.attribute.group.id%' !=
'%ldap.attribute.dn%')",
            "sourcePath": "$.displayName",
            "targetPath": "$.%ldap.attribute.group.id%[0]",
            "targetVariable": "entityIdTargetSystem",
            "scope": "createEntity"
        },
        // If a group is not a direct member of the configured group base
        // path, then its distinguishedName is configured to be equal to CN =
        // <displayName>,<nested_path>,<base_path>,
        // where <nested_path> is read from "sourcePath": "$
        [ 'urn:sap:cloud:scim:schemas:extension:ad:2.0:Group' ][ 'nestedPath' ]"
        {
            "condition": "('%ldap.attribute.group.id%' ==
'%ldap.attribute.dn%')",
            "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:ad:2.0:Group' ][ 'nestedPath' ]",
            "optional": true,
            "targetVariable": "nestedPathVariable",
            "defaultValue": "",
            "scope": "createEntity"
        }
    ]
}

```

```

    {
      "condition": "('%ldap.attribute.group.id%' ==
'%ldap.attribute.dn%')",
      "sourcePath": "$.displayName",
      "functions": [
        {
          "function": "concatString",
          "prefix": "CN="
        },
        {
          "condition": "('${nestedPathVariable}' != '')",
          "function": "concatString",
          "suffix": ", "
        },
        {
          "condition": "('${nestedPathVariable}' != '')",
          "function": "concatString",
          "suffix": "${nestedPathVariable}"
        },
        {
          "function": "concatString",
          "suffix": ",%ldap.group.path%"
        }
      ],
      "targetPath": "$.%ldap.attribute.group.id[0]",
      "targetVariable": "entityIdTargetSystem",
      "scope": "createEntity"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.cn[0]",
      "scope": "createEntity"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.sAMAccountName[0]"
    },
    {
      "sourcePath": "$.members[*]",
      "targetVariable": "membersVariable",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "functions": [
        {
          "condition": "@.type != 'Group'",
          "entityType": "user",
          "applyOnElements": true,
          "type": "resolveEntityIds"
        },
        {
          "condition": "@.type == 'Group'",
          "entityType": "group",
          "applyOnElements": true,
          "type": "resolveEntityIds"
        },
        {
          "condition": "(@.type != 'Group') &&
('%ldap.attribute.user.id%' != '%ldap.attribute.dn%')",
          "function": "concatString",
          "applyOnElements": true,
          "applyOnAttribute": "value",
          "prefix": "%ldap.attribute.user.id%=",
          "suffix": ",%ldap.user.path%"
        },
        {
          "condition": "(@.type == 'Group') &&
('%ldap.attribute.group.id%' != '%ldap.attribute.dn%')",
          "function": "concatString",

```

```

        "applyOnElements": true,
        "applyOnAttribute": "value",
        "prefix": "%ldap.attribute.group.id%",
        "suffix": ",%ldap.group.path%"
    }
  ],
  {
    "sourceVariable": "membersVariable",
    "variablePath": "$[*].value",
    "targetPath": "$.member",
    "defaultValue": [],
    "preserveArrayWithSingleElement": true,
    "optional": true
  }
]
}

```

- Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

### Note

Identity Provisioning escapes the following special characters: comma (,), plus (+) and semicolon (;) in the CN component of the distinguished name (DN). This means that, if a user attribute in the source system contains a special character (**John, Smith**), and this attribute is mapped to the CN in Microsoft AD target system, the comma will be escaped in the DN.

The following special characters cannot be escaped and result in an error: equal sign (=), less than symbol (<), greater than symbol (>), hash sign (#) and backslash (\).

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## Related Information

[Microsoft AD: Technical Documents](#) ➡

[Setting Timeout for Ldap Operations](#) ➡

[Connection Pooling Configuration](#) ➡



## 1.6.2.40 Microsoft Azure Active Directory

Follow this procedure to set up Microsoft Azure Active Directory (in short, Azure AD) as a target system.

### Prerequisites

#### ! Restriction

This system is available for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You've logged on to Microsoft Azure Portal, with credentials for a user with directory role **Global administrator**. For more information, see [Microsoft: Assigning administrator roles in Azure Active Directory](#) ➤.
- In ► [Azure Active Directory](#) ► [App registrations](#) ▾, you've registered an application with a secret key and permissions (see below) for Microsoft Graph API. These permissions must be consented by an administrator. For more information, see [Microsoft Graph permissions reference](#) ➤.
- (Relevant to target systems) Your registered application is assigned the **User Account Administrator** role. This role allows you to deprovision users. For more information, see [MS Azure PowerShell: Add-MsolRole Member](#) ➤.

#### i Note

If this role isn't assigned, you can only disable users. To do that, set the `accountEnabled` property to **false**. For more information, see [MS Graph: user resource type](#) ➤.

### Permissions

Assign the following permissions to your application, according to your scenario. Also, the permissions have to be of type [Application](#).

- Users – [User.ReadWrite.All](#), [Directory.AccessAsUser.All](#)
- Groups – [Group.ReadWrite.All](#)

For more information, see [MS Graph: Users](#) ➤ and [MS Graph: Groups](#) ➤.

### Context

When using it as a target system, you can write both users and groups, read from any source system you've added in the Identity Provisioning user interface. The Azure AD target systems use Microsoft Graph API. For more information, see [Microsoft Graph](#) ➤.

If you've successfully finished with the initial setup (described in the **Prerequisites** section), continue with the procedure.

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *Microsoft Azure Active Directory* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter: <a href="https://graph.microsoft.com">https://graph.microsoft.com</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the application ID registered in your Azure AD subscription (see the <b>Prerequisites</b> section).
Password	Enter the secret key associated to your app registration.
aad.domain.name	Enter one of the verified domain names from the corresponding Azure AD tenant. On this domain, you perform the provisioning operations. For more information, see <a href="#">Microsoft: Manage domain names</a> ➔.
oauth.resource.name	Enter: <a href="https://graph.microsoft.com">https://graph.microsoft.com</a>
OAuth2TokenServiceURL	Enter: <a href="https://login.microsoftonline.com/&lt;your_domain&gt;/oauth2/token">https://login.microsoftonline.com/&lt;your_domain&gt;/oauth2/token</a> , where <your_domain> is the domain name you have set in the <b>aad.domain.name</b> property.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default

transformation for the [Microsoft Azure Active Directory](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in Microsoft Active Directory. For more information, see:

[Manage Transformations \[page 1494\]](#)

[MS Graph: Users](#) ➤

[MS Graph: Groups](#) ➤

**Default transformation:**

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.onPremisesImmutableId",
        "optional": true,
        "targetPath": "$.onPremisesImmutableId"
      },
      {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.accountEnabled"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.mailNickname"
      },
      {
        "sourcePath": "$.displayName",
        "optional": true,
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$.name.givenName",
        "optional": true,
        "targetPath": "$.givenName"
      },
      {
        "sourcePath": "$.name.familyName",
        "optional": true,
        "targetPath": "$.surname"
      },
      {
        "sourcePath": "$.addresses[0].locality",
        "optional": true,
        "targetPath": "$.city"
      },
      {
        "sourcePath": "$.addresses[0].country",
        "optional": true,
        "targetPath": "$.country"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userPrincipalName",
        "scope": "createEntity",
        "functions": [
          {
```

```

        "type": "concatString",
        "suffix": "@%aad.domain.name%"
    }
}
],
{
    "sourcePath": "$.active",
    "targetPath": "$.accountEnabled",
    "scope": "createEntity"
},
{
    "sourcePath": "$.name.givenName",
    "targetPath": "$.mailNickname",
    "scope": "createEntity"
},
{
    "sourcePath": "$.displayName",
    "targetPath": "$.displayName",
    "scope": "createEntity"
},
{
    "targetPath": "$.passwordProfile.password",
    "scope": "createEntity",
    "functions": [
        {
            "type": "randomPassword",
            "passwordLength": 16,
            "minimumNumberOfLowercaseLetters": 1,
            "minimumNumberOfUppercaseLetters": 1,
            "minimumNumberOfDigits": 1,
            "minimumNumberOfSpecialSymbols": 0
        }
    ]
},
{
    "constant": false,
    "targetPath":
"$$.passwordProfile.forceChangePasswordNextSignIn",
    "scope": "createEntity"
}
],
},
"group": {
    "mappings": [
        {
            "sourceVariable": "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "sourcePath": "$.displayName",
            "optional": true,
            "targetPath": "$.displayName"
        },
        {
            "sourcePath": "$.description",
            "optional": true,
            "targetPath": "$.description"
        },
        {
            "sourcePath": "$.allowExternalSenders",
            "optional": true,
            "targetPath": "$.allowExternalSenders"
        },
        {
            "sourcePath": "$.autoSubscribeNewMembers",
            "optional": true,
            "targetPath": "$.autoSubscribeNewMembers"
        }
    ]
},

```

```

    {
      "sourcePath": "$.isSubscribedByMail",
      "optional": true,
      "targetPath": "$.isSubscribedByMail"
    },
    {
      "sourcePath": "$.visibility",
      "optional": true,
      "targetPath": "$.visibility"
    },
    {
      "sourcePath": "$.securityEnabled",
      "optional": true,
      "targetPath": "$.securityEnabled"
    },
    {
      "sourcePath": "$.mailEnabled",
      "optional": true,
      "targetPath": "$.mailEnabled"
    },
    {
      "sourcePath": "$.displayName",
      "targetPath": "$.displayName",
      "scope": "createEntity"
    },
    {
      "sourcePath": "$.externalId",
      "targetPath": "$.mailNickname",
      "scope": "createEntity"
    },
    {
      "constant": true,
      "targetPath": "$.mailEnabled",
      "scope": "createEntity"
    },
    {
      "constant": false,
      "targetPath": "$.securityEnabled",
      "scope": "createEntity"
    },
    {
      "constant": "Unified",
      "targetPath": "$.groupTypes[0]",
      "scope": "createEntity"
    }
  ]
}

```

5. Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.41 SCIM System

Follow this procedure to set up a SCIM system as a target system.

### Prerequisites

#### ! Restriction

This system is available for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need this only if the SCIM system is exposed in a private corporate network. For more information, see [Cloud Connector](#).
- You have technical user credentials for a SCIM system, with read/write access permissions, depending on the scenario you want to implement. In case OAuth is used for authentication, client ID and secret are required when creating a destination for access token retrieval.

### Context

Create a general SCIM 2.0 based target system to write users and groups to it.

### Procedure

1. (Optional) If the SCIM system is exposed in a private corporate network, add an access control system mapping in Cloud Connector. For more information, see [Configure Access Control \(HTTP\)](#).
2. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Add *SCIM System* as a target system. For more information, see [Add a System \[page 1477\]](#).
4. Choose the *Properties* tab to configure the connection settings for your system.

#### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the service URL. For example:  <a href="#">http://&lt;cloudfoundry_server&gt;.com/api/uaa/</a>
ProxyType	Depending on your network exposure, enter one of the following: <ul style="list-style-type: none"><li>• <a href="#">Internet</a></li><li>• <a href="#">OnPremise</a></li></ul>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	You can specify one of the following: <ul style="list-style-type: none"><li>• Technical user ID</li><li>• Client ID for OAuth HTTP destinations. It's used for retrieving of the access token.</li></ul>
Password	You can enter one of the following: <ul style="list-style-type: none"><li>• Technical user password</li><li>• Client secret for OAuth HTTP destinations. It's used for retrieving of the access token.</li></ul>
OAuth2TokenServiceURL	If you need to make OAuth authentication to the system, enter the URL to the access token provider service for OAuth HTTP destinations.  For example: <a href="#">https://&lt;token_provider&gt;.com/api/oauth2/v2.0/token</a>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 5. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SCIM](#) target system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SCIM system. For more information, see [Manage Transformations \[page 1494\]](#).

To make group assignments via the user resource, you need to change the default transformation of the target system as described in [Enabling Group Assignment \[page 1499\]](#).

- **Mapping logic** – The behavior of the default transformation logic is to map all attributes from the internal SCIM representation to the target entity. If the entity has e-mail addresses, the first entry will be marked as primary.
- **User off-boarding** – Users can be deleted from the target system. Depending on the implementation, this could be done through a user interface (if such exists) or the SCIM REST API. Users could be deactivated, depending on the SCIM system implementation. The SCIM core schema defines an attribute “**active**”, whose definition depends on the service provider. For more information, see [SCIM: Singular Attributes](#) ➔

#### Default transformation:

##### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "targetPath": "$.id",
        "type": "remove"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "condition": "$.emails[0].length() > 0",
        "targetPath": "$.emails[0].primary",
        "constant": true
      },
      {
        "type": "remove",
        "targetPath": "$"
      }
    ],
    [ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
  },
  "group": {
    "ignore": true,
    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "targetPath": "$.id",
        "type": "remove"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "targetPath": "$.members",
        "type": "remove"
      },
      {
        "sourcePath": "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$.members[?(@.value)]",
      }
    ]
  }
}
```



```

    "functions": [
      {
        "type": "resolveEntityIds"
      }
    ]
  }
}

```

- Now, add a source system from which to read users and groups. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

- Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
- Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.2.42 SSH Server (Beta)

Follow this procedure to set up an SSH server (Beta) as a target system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Identity Access Governance** bundle option.

- You have credentials for a tenant in SAP Business Technology Platform. For more information, see: [Accounts](#)
- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need this only when your SSH server resides in a remote system, outside your Neo environment. For more information, see [Cloud Connector](#).

#### i Note

This is a beta feature available on SAP Business Technology Platform. For more information, see: *Enable beta features* in [Change Subaccount Details](#)

## Context

**SSH Server** is a system (connector) in beta state. It helps you execute bash scripts through SSH connection. The configuration allows you to attach separate scripts per entity lifecycle callback (such as user create, group create/update, and so on). This system helps you connect to remote machines via SSH tunnel, with or without use of the Cloud Connector, depending on whether the SSH port is visible or not.

The bash scripts can take as parameters fields that are coming from the entity JSON data. For example: `sudo su - vcap /home/myscript.sh $.userName $.email`

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Add *SSH Server (Beta)* as a target system. For more information, see [Add a System \[page 1477\]](#).
3. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

Below are listed all available SSH Server properties. Some of them can be mandatory and others – optional, depending on your scenario.

Mandatory Properties

Property Name	Description & Value
ProxyType	Possible values: <ul style="list-style-type: none"><li>• <b>Internet</b> – if the SSH port is visible in your Neo environment</li><li>• <b>OnPremise</b> – if the SSH port is not directly accessible, and you have to use the Cloud Connector. You have to configure TCP protocol connection to the SSH host and port (specify the configuration properties <code>ssh.host</code> and <code>ssh.port</code>).</li></ul>

Property Name	Description & Value
CloudConnectorLocationId	Relevant when the proxy type is <i>OnPremise</i> . Use it only if your SAP Business Technology Platform account uses more than one Cloud Connector.
ssh.create.user.command	Path to the bash command you need to execute to create a user.
ssh.update.user.command	Path to the bash command you need to execute to update a user.
ssh.delete.user.command	Path to the bash command you need to execute to delete a user.
ssh.create.group.command	Path to the bash command you need to execute to create a group.
ssh.update.group.command	Path to the bash command you need to execute to update a group.
ssh.delete.group.command	Path to the bash command you need to execute to delete a group.
ssh.create.user.command.exit.code.already.exists	Integer number (code). It's generated by the "create user" command when the provisioning job tries to create a user that already exists in the target system.
ssh.update.user.command.exit.code.not.found	Integer number (code). It's generated by the "update user" command when the provisioning job tries to update a user that is missing from the target system.
ssh.delete.user.command.exit.code.not.found	Integer number (code). It's generated by the "delete user" command when the provisioning job tries to delete a user that is missing from the target system.
ssh.create.group.command.exit.code.already.exists	Integer number (code). It's generated by the "create group" command when the provisioning job tries to create a group that already exists in the target system.
ssh.update.group.command.exit.code.not.found	Integer number (code). It's generated by the "update group" command when the provisioning job tries to update a group that is missing from the target system.
ssh.delete.group.command.exit.code.not.found	Integer number (code). It's generated by the "delete group" command when the provisioning job tries to delete a group that is missing from the target system.

Property Name	Description & Value
ssh.auth.type	Supported SSH authentication types: <ul style="list-style-type: none"> <li>• <b>key</b></li> <li>• <b>pwd</b></li> <li>• <b>otp</b></li> <li>• <b>key+otp</b></li> <li>• <b>key+pwd</b></li> <li>• <b>pwd+otp</b></li> <li>• <b>key+pwd+otp</b></li> </ul>
ssh.host	
ssh.port	22
ssh.username	
ssh.password	(Credential) Taken into account only if the authentication type includes <b>pwd</b> . That means any of the following: <ul style="list-style-type: none"> <li>• ssh.auth.type = <i>pwd</i></li> <li>• ssh.auth.type = <i>pwd+otp</i></li> <li>• ssh.auth.type = <i>key+pwd</i></li> <li>• ssh.auth.type = <i>key+pwd+otp</i></li> </ul>
ssh.totp.secret.key	(Credential) Taken into account only if the authentication type includes <b>otp</b> . That means any of the following: <ul style="list-style-type: none"> <li>• ssh.auth.type = <i>otp</i></li> <li>• ssh.auth.type = <i>key+otp</i></li> <li>• ssh.auth.type = <i>pwd+otp</i></li> <li>• ssh.auth.type = <i>key+pwd+otp</i></li> </ul>
ssh.private.key.type	The type of SSH private key. Possible values: <ul style="list-style-type: none"> <li>• <b>ssh-rsa</b></li> <li>• <b>ssh-dsa</b></li> </ul> Default value: <i>ssh-rsa</i> <div> <b>i Note</b>  If you choose <i>ssh-rsa</i>, the key should be in format <b>PKCS #8</b>, non-encrypted. </div>
ssh.private.key	(Credential) Taken into account only if the authentication type includes <b>key</b> . That means any of the following: <ul style="list-style-type: none"> <li>• ssh.auth.type = <i>key</i></li> <li>• ssh.auth.type = <i>key+pwd</i></li> <li>• ssh.auth.type = <i>key+otp</i></li> <li>• ssh.auth.type = <i>key+pwd+otp</i></li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

4. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SSH Server (Beta)* target system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SSH server. For more information, see [Manage Transformations \[page 1494\]](#).

**Default transformation:**

≡ Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      }
    ]
  }
}
```

5. Now, add a source system from which to read users. Choose from: [Source Systems \[page 452\]](#)

## Next Steps

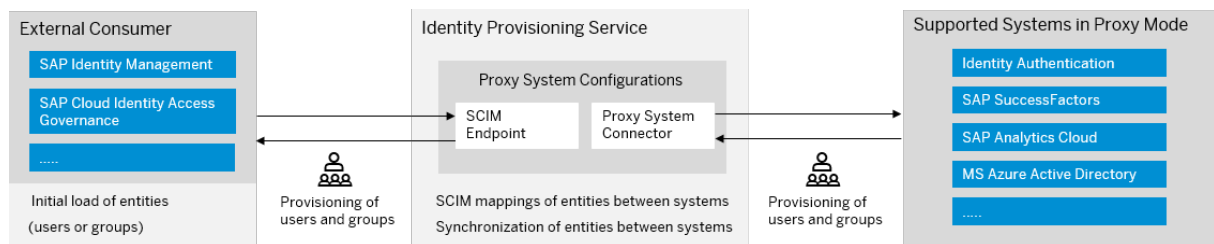
1. Before starting a provisioning job, you can first subscribe for e-mail notifications from the source system you use in your scenario. This way, you will be notified by e-mail about eventual failed entities during the jobs. For more information, see [Manage Job Notifications \[page 1605\]](#).
2. Now, start an identity provisioning job. For more information, see [Monitor Provisioning Job Logs \[page 1594\]](#).

## 1.6.3 Proxy Systems

A proxy system is a special connector type you can use for *hybrid* scenarios.

It exposes any Identity Provisioning supported backend system as a SCIM 2.0 service provider, which can be consumed by any [SCIM 2.0](#) compatible client application, without making a direct connection between them.

After the proxy connector is configured, a consumer application can start sending CRUD requests to the Identity Provisioning proxy connector (which will play the role of a target system). The proxy connector will then read the entities (playing the role of a source system) and provision them to the back-end of the SCIM 2.0 system.



## How Proxy Systems Work

The Identity Provisioning service exposes a SCIM 2.0 based system (connector) as a proxy. Then, an external consumer system regards this proxy connector as its back-end system. Depending on the infrastructure/ environment your Identity Provisioning tenant (bundle or standalone) runs on, you need to perform the following steps to start using a proxy system:

### Infrastructure of SAP Cloud Identity Services

Your tenant type meets one of the following requirements:

- Your bundle tenant is created after March 15, 2022.
  - Your Identity Provisioning is purchased as a standalone product between September 1, 2020 and October 20, 2020.
1. Sign in to the SAP Cloud Identity Services administration console and navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
  2. Choose your technical user (administrator user of type **System**). If you don't have a technical user yet, create one. For more information, see [Add System as Administrator](#)
  3. Configure the technical user authentication.
    - **Certificate**  
Choose ► [Certificate](#) ► [Configure certificate](#) ► and [Upload](#) your certificate.  
Client certificates are used by HTTP REST clients as a means for SSL certificate authentication.
    - **Secrets**  
Choose ► [Secrets](#) ► [Add](#) ► and provide the required information. After saving it, a Client ID and Client Secret are generated for the technical user. Make sure you copy and save the client secret.
  4. Enable the [Access Proxy System API](#) permission for the technical user.
  5. Then navigate to ► [Identity Provisioning](#) ► [Proxy Systems](#) ►, and create a proxy system.
  6. (Optional) Export this system as a [.csv](#) file. This will help an administrator of the external consumer application to import the proxy configuration as a SCIM repository in SAP Identity Management, for example.

#### i Note

The entities exposed by the back-end system will be mapped to SCIM 2.0 entities, if possible. If not possible, the SCIM standard provides a mechanism to define a new resource type with the appropriate schema. You can use the custom resource type to map the back-end entities.

7. Finally, the external application can start sending REST web service requests to the proxy system in order to read identities from the back end of the SCIM 2.0 system. For the authentication, you need to use the [user ID](#) and [password](#) of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).

## SAP BTP, Neo environment

Your tenant type meets one of the following requirements:

- Your bundle tenant is created before March 15, 2022.
  - Your Identity Provisioning is purchased as a standalone product before September 1, 2020.
1. Create a technical user that will be used to connect to the Identity Provisioning proxy system and assign the necessary authorizations to your technical user.
    - For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*
    - For **OAuth authentication**, proceed as follows:  
Step **a.** and **b.** are relevant for Identity Provisioning bundle tenants. For standalone tenants, proceed from step **c.**
      1. Open the Identity Provisioning admin console and choose section ► [Security](#) ► [Authorizations](#) ►.
      2. Select your admin user and set role [Manage OAuth Clients](#) to it. You can grant additional administrators with this role.
      3. Go to the SAP BTP cockpit → [Neo](#) → [Overview](#) and open your subaccount.

### i Note

You can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

4. Register a new OAuth client for the subscription to the [ipsproxy](#) application.
  5. Assign role IPS\_PROXY\_USER to the [oauth\\_client\\_<client\\_ID>](#).
2. Then open again the Identity Provisioning admin console, and create a proxy system.
  3. (Optional) Export this system as a [.csv](#) file. This will help an administrator of the external consumer application to import the proxy configuration as a SCIM repository in SAP Identity Management, for example.

### i Note

The entities exposed by the back-end system will be mapped to SCIM 2.0 entities, if possible. If not possible, the SCIM standard provides a mechanism to define a new resource type with the appropriate schema. You can use the custom resource type to map the back-end entities.

4. Finally, the external application can start sending REST web service requests to the proxy system in order to read identities from the back end of the SCIM 2.0 system. For the authentication, you need to use the [client ID](#) and [secret](#) of the registered OAuth client for which you have assigned the IPS\_PROXY\_USER role.

## How Proxy Transformations Work

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users* or */Groups*) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

### ❖ Example

#### Conditions in Proxy Scenarios

Using conditions is supported for both - proxy *Read Transformation* and proxy *Write Transformation*. However, when conditions are applied to users or groups in proxy *Read Transformation*, the number of returned resources may be "0" or less than the actual number of read entities. This is because some of the entities are filtered out as they don't match the applied condition.

In the example below, the returned resources are "0" because all 5 users (items) returned per page are filtered out as they don't match a condition.

```
SCIM proxy client request: GET /Users?startIndex=6&count=5
SCIM proxy application response:
{
  "startIndex": 6,
  "itemsPerPage": 5,
  "totalResults": 11,
  "Resources": [],
  "schemas": [ "urn:ietf:params:scim:api:messages:2.0:ListResponse" ]
}
```

## How to call a proxy system

As proxy operations cannot be maintained by the Identity Provisioning UI, you need to manage resources (users, groups, schemas) by sending SCIM 2.0 API requests to certain endpoints. Below are listed all endpoints and operations available in the Identity Provisioning service. Each provisioning system, however, supports only a specific set of operations.

**Tip:** The ID of each proxy system (*system\_ID* in the table below) is a dash-separated string. You can see it at the end of the system URL in the Identity Provisioning UI.

### ⚠ Caution

Effective *September 2020*, Shanghai (China) tenants that reside on SAP BTP, Neo environment should be accessed on the following domain: **dispatcher.cn1.platform.sapcloud.cn**



So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

Depending on the infrastructure/environment your Identity Provisioning tenant (bundle or standalone) runs on, use the relevant URI patterns to call an endpoint:

Resource Operation	HTTP Method	Endpoint
Read users	GET	<p><b>SAP BTP, Neo environment</b></p> <ul style="list-style-type: none"><li>• <b>OAuth Authentication:</b> <code>https://ipsproxy&lt;provider_subaccount&gt;-&lt;consumer_subaccount&gt;.&lt;host&gt;/ipsproxy/api/v1/scim/&lt;system_id&gt;/Users</code></li><li>• <b>Certificate Authentication:</b> <code>https://ipsproxy&lt;provider_subaccount&gt;-&lt;consumer_subaccount&gt;.cert.&lt;host&gt;/ipsproxy/certapi/v1/scim/&lt;system-id&gt;/Users</code></li></ul> <hr/> <p><b>Infrastructure of SAP Cloud Identity Services</b></p> <ul style="list-style-type: none"><li>• <b>Basic Authentication and Certificate Authentication:</b> <code>https://&lt;ias-tenant-host&gt;/ipsproxy/service/api/v1/scim/&lt;system_id&gt;/Users</code></li></ul>

Resource Operation	HTTP Method	Endpoint
Read a user	GET	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li> <b>OAuth Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/Users/  &lt;user_id&gt;</code> </li> <li> <b>Certificate Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Users/  &lt;user_id&gt;</code> </li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li> <b>Basic Authentication and Certificate Authentication:</b> <code>https://&lt;ias-tenant-host&gt;/ipsproxy/  service/api/v1/scim/  &lt;system_id&gt;/Users/  &lt;user_id&gt;</code> </li> </ul>

Resource Operation	HTTP Method	Endpoint
Create a user	POST	<p><b>SAP BTP, Neo environment</b></p> <ul style="list-style-type: none"> <li>• <b>OAuth Authentication:</b>  https://  ipsproxy&lt;provider_subaccount&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/Users</li> <li>• <b>Certificate Authentication:</b>  https://  ipsproxy&lt;provider_subaccount&gt;-  &lt;consumer_subaccount&gt;.&lt;  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Users</li> </ul> <hr/> <p><b>Infrastructure of SAP Cloud Identity Services</b></p> <ul style="list-style-type: none"> <li>• <b>Basic Authentication and Certificate Authentication:</b> https://&lt;ias-tenant-host&gt;/ipsproxy/service/api/v1/scim/&lt;system_id&gt;/Users</li> </ul>

Resource Operation	HTTP Method	Endpoint
Update a user ( <i>full</i> )	PUT	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li> <b>OAuth Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/Users/  &lt;user_id&gt;</code> </li> <li> <b>Certificate Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Users/  &lt;user_id&gt;</code> </li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li> <b>Basic Authentication and Certificate Authentication:</b> <code>https://&lt;ias-tenant-host&gt;/ipsproxy/  service/api/v1/scim/  &lt;system_id&gt;/Users/  &lt;user_id&gt;</code> </li> </ul>

Resource Operation	HTTP Method	Endpoint
Update a user ( <i>partial</i> )	PATCH	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li> <b>OAuth Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/Users/  &lt;user_id&gt;</code> </li> <li> <b>Certificate Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Users/  &lt;user_id&gt;</code> </li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li> <b>Basic Authentication and Certificate Authentication:</b> <code>https://&lt;ias-tenant-host&gt;/ipsproxy/service/api/v1/scim/&lt;system_id&gt;/Users/&lt;user_id&gt;</code> </li> </ul>

Resource Operation	HTTP Method	Endpoint
Delete a user	DELETE	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li> <b>OAuth Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/Users/  &lt;user_id&gt;</code> </li> <li> <b>Certificate Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Users/  &lt;user_id&gt;</code> </li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li> <b>Basic Authentication and Certificate Authentication:</b> <code>https://&lt;ias-tenant-host&gt;/ipsproxy/  service/api/v1/scim/  &lt;system_id&gt;/Users/  &lt;user_id&gt;</code> </li> </ul>

Resource Operation	HTTP Method	Endpoint
Read groups	GET	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li> <b>OAuth Authentication:</b>  https://  ipsproxy&lt;provider_subaccount&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/Groups </li> <li> <b>Certificate Authentication:</b>  https://  ipsproxy&lt;provider_subaccount&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Groups </li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li> <b>Basic Authentication and Certificate Authentication:</b> https://&lt;ias-tenant-host&gt;/ipsproxy/  service/api/v1/scim/  &lt;system_id&gt;/Groups </li> </ul>

Resource Operation	HTTP Method	Endpoint
Read a group	GET	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li>• <b>OAuth Authentication:</b>  https://  ipsproxy&lt;provider_subaccount&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/  Groups/&lt;group_id&gt;</li> <li>• <b>Certificate Authentication:</b>  https://  ipsproxy&lt;provider_subaccount&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Groups/  &lt;group_id&gt;</li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li>• <b>Basic Authentication and Certificate Authentication:</b> https://&lt;ias-tenant-host&gt;/ipsproxy/  service/api/v1/scim/  &lt;system_id&gt;/Groups/  &lt;group_id&gt;</li> </ul>



Resource Operation	HTTP Method	Endpoint
Create a group	POST	<p><b>SAP BTP, Neo environment</b></p> <ul style="list-style-type: none"> <li> <b>OAuth Authentication:</b>  https://  ipsproxy&lt;provider_subaccount&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/Groups </li> <li> <b>Certificate Authentication:</b>  https://  ipsproxy&lt;provider_subaccount&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Groups </li> </ul> <hr/> <p><b>Infrastructure of SAP Cloud Identity Services</b></p> <ul style="list-style-type: none"> <li> <b>Basic Authentication and Certificate Authentication:</b> https://&lt;ias-tenant-host&gt;/ipsproxy/  service/api/v1/scim/  &lt;system_id&gt;/Groups </li> </ul>

Resource Operation	HTTP Method	Endpoint
Update a group ( <i>full</i> )	PUT	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li> <b>OAuth Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/  Groups/&lt;group_id&gt;</code> </li> <li> <b>Certificate Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Groups/  &lt;group_id&gt;</code> </li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li> <b>Basic Authentication and Certificate Authentication:</b> <code>https://&lt;ias-tenant-host&gt;/ipsproxy/  service/api/v1/scim/  &lt;system_id&gt;/Groups/  &lt;group_id&gt;</code> </li> </ul>

Resource Operation	HTTP Method	Endpoint
Update a group ( <i>partial</i> )	PATCH	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li> <b>OAuth Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/  Groups/&lt;group_id&gt;</code> </li> <li> <b>Certificate Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Groups/  &lt;group_id&gt;</code> </li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li> <b>Basic Authentication and Certificate Authentication:</b> <code>https://&lt;ias-tenant-host&gt;/ipsproxy/  service/api/v1/scim/  &lt;system_id&gt;/Groups/  &lt;group_id&gt;</code> </li> </ul>

Resource Operation	HTTP Method	Endpoint
Delete a group	DELETE	<p>SAP BTP, Neo environment</p> <ul style="list-style-type: none"> <li>• <b>OAuth Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.&lt;  host&gt;/ipsproxy/api/v1/  scim/&lt;system_id&gt;/  Groups/&lt;group_id&gt;</code> </li> <li>• <b>Certificate Authentication:</b>  <code>https://  ipsproxy&lt;provider_subac  count&gt;-  &lt;consumer_subaccount&gt;.c  ert.&lt;host&gt;/ipsproxy/  certapi/v1/scim/  &lt;system-id&gt;/Groups/  &lt;group_id&gt;</code> </li> </ul> <hr/> <p>Infrastructure of SAP Cloud Identity Services</p> <ul style="list-style-type: none"> <li>• <b>Basic Authentication and Certificate Authentication:</b> <code>https://&lt;ias-tenant-host&gt;/ipsproxy/service/api/v1/scim/&lt;system_id&gt;/Groups/&lt;group_id&gt;</code></li> </ul>

## Query Parameters for Proxy SCIM API

When using Identity Provisioning Proxy SCIM API, you can specify the `attributes` and the `excludedAttributes` query parameters to control which user or group resource attributes to be included or excluded from the response.

### **i** Note

The query parameters work for first-level attributes only.

### Code Syntax

In this example, "name" is the first-level attribute. The "familyName", "givenName" and "middleName" are second-level attributes and cannot be used as the query parameter value.

```
"name": {
  "familyName": "Armstrong",
  "givenName": "Julie",
  "middleName": "Grace"
},
```

The query parameter value is the resource attribute name. In case you want to specify multiple attribute names, you need to separate them by comma. Using " " (space) or "," (comma) is not a valid value and results in returning all the resource attributes.

In order for an attribute to be included or excluded from the response, the attribute's schema must be defined in the [Schemas](#) attribute of the user resource. For example, if you want to return all users with the custom attribute [roomNumber](#) in the response, the custom schema must be defined in the [Schemas](#) attribute of the user resource.

The `attributes` and the `excludedAttributes` query parameters can be combined with other parameters, such as: filtering, paging of resources, paging of multi-value attributes.

- `attributes` - When specified, this query parameter defines which user or group resource attributes to be included in the response.

This request example returns all users with the [active](#) and [userName](#) attributes in the response.

### Code Syntax

```
GET /Users?attributes=active,userName
```

This request example returns all groups with the [displayName](#) attribute in the response.

### Code Syntax

```
GET /Groups?attributes=displayName
```

This request example combines the [attributes](#) query parameter with filtering of a user by [userName](#). It returns only the TestUser with both attributes: [active](#) and [displayName](#) in the response.

### Code Syntax

```
GET /Users?filter=userName eq "TestUser"&attributes=active,displayName
```

In case an attribute is defined in two schemas (for example, the [emails](#) attribute is defined in the core schema and in the custom schema), you need to specify the schema URI and the attribute name as the query parameter value. This way, the users will be returned with emails attribute from the specified schema. Otherwise, if you only specify [emails](#) without the schema URI, the [emails](#) attribute from both schemas will be returned.

### Code Syntax

```
GET /Users?
attributes=urn:ietf:params:scim:schemas:extension:sap:2.0:User:emails
```

- `excludedAttributes` - When specified, this query parameter defines which user or group resource attributes to be excluded in the response.  
This request example returns all users with all attributes excluding `userType` in the response.

#### Code Syntax

```
GET /Users?excludedAttributes=userType
```

This request example returns all groups with all attributes excluding `schemas` in the response.

#### Code Syntax

```
GET /Groups?excludedAttributes=schemas
```

This request example combines the `excludedAttributes` query parameter with paging parameters `startIndex` and `count` of users. It returns 3 users starting with the first one as the first query result with all attributes excluding `schemas` in the response.

#### Code Syntax

```
GET /Users?startIndex=1&count=3&excludedAttributes=schemas
```

## Additional Information

Learn more about system types: [System Types \[page 86\]](#)

Learn more about SCIM 2.0 protocol: [SCIM protocol](#) ➔

Proxy scenario example: [Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

SAP Community Blog: [Hybrid Scenarios with Identity Provisioning Proxy](#) ➔

### 1.6.3.1 Identity Authentication

Follow this procedure to set up SAP Cloud Identity Service – Identity Authentication as a proxy system.

## Prerequisites

To establish the connection between Identity Provisioning and Identity Authentication, you need to set up the technical user (of type `System`) in Identity Authentication and assign this user the necessary authorizations. You can do it now (as a prerequisite) or in the process of configuring Identity Authentication as a proxy system, as described in step 5.

## i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

Identity Authentication provides authentication and single sign-on for users in the cloud.

You can use the Identity Provisioning user interface (UI) to configure Identity Authentication as a proxy system for hybrid integration, where Identity Authentication is connected to an external identity management system for user and group provisioning. Using the Identity Provisioning you can read corporate users from the external system and provision them to the Identity Authentication user store (and the other way around) without making a direct connection between these systems. This way, you can implement secure authentication, single sign-on (SSO), strong authentication, and mobile SSO so that the provisioned users to Identity Authentication have access to the business applications of your company.

The proxy systems consume SCIM 2.0 API which is provided by Identity Authentication.

There are two versions of the Identity Authentication SCIM API. They are handled by the `ias.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on July 9, 2021) - Identity Authentication SCIM API (in short, SCIM API version 1) is used. For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#)
- When the value is set to **2** - Identity Directory SCIM API (in short, SCIM API version 2) is used. This value is set automatically for all manually created systems in the Identity Provisioning UI after versioning was introduced on July 9, 2021.

SCIM API version 2 is enhanced to support patch operations for proxy systems only, paging for group members and user's groups, custom attributes, delta read mode for users. Also, the group resource mapping in the transformation is not ignored by default, as it is in SCIM API version 1.

To create Identity Authentication as a proxy system, proceed as follows:

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

## i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <i>Users &amp; Authorizations</i> ► <i>Administrators</i> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <i>Access Proxy System API</i> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <i>Security</i> ► <i>OAuth</i> ► <i>Clients</i> ► and choose <i>Register New Client</i>.</li> <li>2. From the <i>Subscription</i> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <i>Authorization Grant</i> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <i>Secret</i> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <i>Client ID</i>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <i>Subscriptions</i> ► <i>Java Applications</i> ► <i>ipsproxy</i> ►.</li> <li>7. From the left-side navigation, choose ► <i>Roles</i> ► <i>IPS_PROXY_USER</i> ►.</li> <li>8. Choose <i>Assign</i> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *Identity Authentication* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Set up the communication between Identity Provisioning and Identity Authentication and configure your authentication method (basic or certificate-based).

### **i** Note

We recommend that you use certificate-based authentication.

- a. In your newly added Identity Authentication proxy system, select the *Certificate* tab and choose ► *Generate* ► *Download* ►, as described in [Manage Certificates \[page 1506\]](#).

Skip step **a.** if you want to use basic authentication.

In SAP Cloud Identity Services administration console, perform the next steps. They are relevant for both basic and certificate-based authentication.

- b. [Add System as administrator](#) and provide the respective credentials.



For basic authentication, provide a password. The user ID will be generated automatically when you set the password for the first time.

For certificate-based authentication, upload the certificate you have generated in SAP Cloud Identity Services administration console on the previous step.

- c. Save your changes.
  - d. Make sure [Manage Users](#) and [Manage Groups](#) authorization roles are enabled for the technical user. This way, you can create, edit and delete users and groups in the Identity Authentication user store.
6. Choose the [Properties](#) tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL of the Identity Authentication tenant of your company.</p> <p>For example: <a href="https://mytenant.accounts.on-demand.com">https://mytenant.accounts.on-demand.com</a></p> <div><b>i</b> Note If your Identity Authentication Shanghai (China) tenants reside on SAP BTP, Neo environment, you should use the following URL pattern: <a href="https://&lt;tenant_ID&gt;.accounts.sapcloud.cn/">https://&lt;tenant_ID&gt;.accounts.sapcloud.cn/</a></div>
ProxyType	<p>Enter: <a href="#">Internet</a></p> <p>The Identity Authentication is a cloud solution and is outside of your company on-premise infrastructure.</p>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>

Property Name	Description & Value
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the Client ID (previously User ID) of the Identity Authentication technical user. It is generated automatically for the administrator of type system, when choosing <a href="#">Secrets</a> <a href="#">Add</a> <a href="#">Save</a>. For example: <a href="#">1ab7c243-5de5-4530-8g14-1234h26373ab</a></p> <p>If your technical user was created before <a href="#">January 2020</a>, enter the T-user. For example: <a href="#">T000003</a></p>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the Client Secret (previously Password) of the Identity Authentication technical user. It is generated automatically for the administrator of type system, when choosing <a href="#">Secrets</a> <a href="#">Add</a> <a href="#">Save</a>.</p>
Optional Properties	
Property Name	Description & Value
<ul style="list-style-type: none"> <li>ias.&lt;property_name&gt;</li> <li>scim.&lt;property_name&gt;</li> </ul>	<p>When using SCIM API version 2, property names start with <b>ias</b> prefix, for example: <code>ias.user.filter</code>.</p> <p>When using SCIM API version 1, property names start with <b>scim</b> prefix, for example: <code>scim.user.filter</code>.</p> <p>For more information, see <a href="#">List of Properties [page 94]</a>. Use the main search or filter properties by <a href="#">Name</a> or <a href="#">System Type</a> columns.</p>
ias.user.filter	<p>When specified, only those users matching the filter expression will be read.</p> <p>For example: <b>name.familyName eq "Smith" and addresses.country eq "US"</b></p> <p>This filter will read only users whose family name is <a href="#">"Smith"</a> and are living in the <a href="#">United States</a>.</p> <p>For more information, see <a href="#">Identity Directory SCIM API: User Search</a>.</p>

Property Name	Description & Value
<code>ias.group.filter</code>	<p>When specified, only those groups matching the filter expression will be read.</p> <p>For example: <b><code>displayName eq "ProjectTeam1"</code></b></p> <p>This filter will read only groups, whose display name is <i>"ProjectTeam1"</i>.</p> <p>For more information, see <a href="#">Identity Directory SCIM API: Group Search</a>.</p>

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#) standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports *native read filtering*, the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with *'totalResults'* set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type *'tooMany'*.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example: *GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.emails[0].value",
  "preserveArrayWithSingleElement": true,
  "targetPath": "$.emails[0].value"
},
```

You can also set the following filter in the [Properties](#) tab: `scim.user.filter = name.familyName eq "Smith"`

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=emails[0].value eq "john.smith03@dummymail.com"**

The query request to the Identity Provisioning API will result into: **/Users?filter=name.familyName eq "Smith" and emails[0].value eq "john.smith03@dummymail.com"**

7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Identity Authentication* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your Identity Authentication system. For more information, see: [Manage Transformations \[page 1494\]](#)

Identity Authentication: SCIM REST API [Identity Authentication: SCIM REST API](#)

SCIM API version 2: [Identity Directory SCIM API](#) 

Default read and write transformations:

→ Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users* or */Groups*) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Default transformations for SCIM API version 1:

### Read Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$ .userUuid",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ][ 'userUuid' ]"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "optional": true,
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$ .emails[*].value",
        "preserveArrayWithSingleElement":
true,
        "targetPath": "$.emails[?
(@.value)]"
      }
    ]
  }
}
```

### Write Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "condition":
"($.emails.length() > 0) &&
($.name.familyName EMPTY false)",
    "mappings": [
      {
        "sourcePath": "$.groups",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$ .corporateGroups"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:User",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:exte
nsion:enterprise:2.0:User",
        "targetPath":
"$ .schemas[1]"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "optional": true
      },
      {
        "sourcePath":
"$ .emails[*].value",
        "preserveArrayWithSingleElement":
true,
        "targetPath": "$.emails[?
(@.value)]"
      },
      {
        "sourcePath":
"$ .userType",
        "targetPath":
"$ .userType",
        "optional": true
      }
    ]
  }
}
```

```

    },
    {
      "sourcePath":
"$$.emails[0].value",
      "targetPath":
"$$.emails[0].value"
    },
    {
      "sourcePath": "$$.emails[?
(@.primary== true)].value",
      "correlationAttribute":
true
    },
    {
      "sourcePath": "$$.active",
      "targetPath": "$$.active"
    },
    {
      "sourcePath":
"$$.userType",
      "targetPath":
"$$.userType",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.givenName",
      "targetPath":
"$$.name.givenName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.middleName",
      "targetPath":
"$$.name.middleName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.familyName",
      "targetPath":
"$$.name.familyName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "targetPath":
"$$.name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$$.addresses",
      "targetPath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
  },

```

```

    },
    {
      "sourcePath":
"$$.name.givenName",
      "targetPath":
"$$.name.givenName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.middleName",
      "targetPath":
"$$.name.middleName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.familyName",
      "targetPath":
"$$.name.familyName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "targetPath":
"$$.name.honorificPrefix",
      "optional": true
    },
  },
  // If a user address misses type,
  the putIfAbsent function sets the
  default address type - work.
  // If a user address has type
  but it's different than home or
  work, the putIfPresent function
  sets the default address type -
  work.
  // If a user address is of type
  home or work, the putIfPresent
  function keeps this value as is.
  {
    "sourcePath":
"$$.addresses",
    "targetPath":
"$$.addresses",
    "preserveArrayWithSingleElement":
true,
    "defaultValue": [],
    "optional": true,
    "functions": [
      {
        "function":
"putIfAbsent",
        "key": "type",
        "defaultValue": "work"
      },
      {
        "condition": "(@.type
NIN ['work', 'home'])",
        "function":
"putIfPresent",

```

```

    {
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "optional": true
    },
    {
      "sourcePath":
        "$.phoneNumbers",
      "targetPath":
        "$.phoneNumbers",
      "preserveArrayWithSingleElement":
        true,
      "optional": true
    },
    {
      "sourcePath":
        "$.timezone",
      "targetPath":
        "$.timezone",
      "optional": true
    },
    {
      "sourcePath":
        "$.displayName",
      "targetPath":
        "$.displayName",
      "optional": true
    },
    {
      "sourcePath":
        "$.sourceSystem",
      "targetPath":
        "$.sourceSystem",
      "ignore": true
    },
    {
      "sourcePath": "$.groups",
      "targetPath": "$.groups",
      "preserveArrayWithSingleElement":
        true,
      "optional": true
    },
    {
      "type": "remove",
      "targetPath":
        "$.groups[*].display",
      "condition":
        "$.displayName EMPTY true",
      "type": "remove",
      "targetPath":
        "$.displayName"
    },
    {
      "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",

```

```

      "key": "type",
      "defaultValue": "work"
    }
  ]
},
{
  "sourcePath": "$.locale",
  "targetPath": "$.locale",
  "optional": true
},
{
  "sourcePath":
    "$.phoneNumbers",
  "targetPath":
    "$.phoneNumbers",
  "preserveArrayWithSingleElement":
    true,
    "optional": true
  },
  {
    "sourcePath":
      "$.displayName",
      "targetPath":
        "$.displayName",
      "optional": true
    },
    {
      "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
      "optional": true
    },
    {
      "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'costCenter' ]",
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'costCenter' ]",
      "optional": true
    },
    {
      "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'organization' ]",
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'organization' ]",
      "optional": true
    },
    {
      "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext

```

```

        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
        "optional": true
    },
    {

```

```

        "targetPath": "$
['division']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['displayName']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "defaultValue": true,
        "optional": true
    },
    {
        "constant": "false",
        "targetPath":
        "$.sendMail",
        "scope": "createEntity"
    },
    {
        "constant": "true",
        "targetPath":
        "$.mailVerified",
        "scope": "createEntity"
    },
    // There will be no initial
    password provided by default.
    That's why passwordStatus is
    disabled.

```



```

        "sourcePath": "$
        ['urn:ietf:params:scim:schemas:ext
        ension:enterprise:2.0:User']
        ['manager']['displayName']",
        "targetPath": "$
        ['urn:ietf:params:scim:schemas:ext
        ension:enterprise:2.0:User']
        ['manager']['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$
        ['urn:sap:cloud:scim:schemas:exten
        sion:custom:2.0:User']",
        "targetPath": "$
        ['urn:sap:cloud:scim:schemas:exten
        sion:custom:2.0:User']",
        "optional": true
    },
    {
        "sourcePath": "$.company",
        "targetPath": "$
        ['urn:ietf:params:scim:schemas:ext
        ension:enterprise:2.0:User']
        ['organization']",
        "optional": true
    }
],
"group": {
    "scimEntityEndpoint":
    "Groups",
    "mappings": [
        {
            "sourcePath": "$.id",
            "targetPath": "$.id",
            "targetVariable":
            "entityIdSourceSystem"
        },
        {
            "sourceVariable":
            "entityBaseLocation",
            "targetVariable":
            "entityLocationSourceSystem",
            "targetPath":
            "$.meta.location",
            "functions": [
                {
                    "type":
                    "concatString",
                    "suffix": "$
                    {entityIdSourceSystem}"
                }
            ]
        }
    ],
    "constant":
    "urn:ietf:params:scim:schemas:core
    :2.0:Group",
    "targetPath":
    "$.schemas[0]"
},

```

```

    {
        "constant": "disabled",
        "targetPath":
        "$.passwordStatus",
        "scope": "createEntity"
    },
    // The sourceSystem attribute
    shows the provisioning source of
    the users. The supported value is
    39.
    // That means, a corporate user
    is provisioned via the Identity
    Authentication REST API. See the
    Remember box below.
    {
        "constant": "39",
        "targetPath":
        "$.sourceSystem",
        "scope": "createEntity"
    },
    // The userType attribute accepts
    different values. The default one
    is employee.
    // If you set it to public, that
    means Identity Authentication is
    the default password store. See
    the Remember box below.
    {
        "constant": "employee",
        "targetPath": "$.userType"
    },
    {
        "sourcePath":
        "$.timezone",
        "targetPath":
        "$.timeZone",
        "optional": true
    }
],
"group": {
    "scimEntityEndpoint":
    "Groups",
    "mappings": [
        {
            "sourceVariable":
            "entityIdTargetSystem",
            "targetPath": "$.id"
        },
        {
            "sourcePath":
            "$.displayName",
            "targetPath":
            "$.displayName"
        },
        {
            "scope": "createEntity",
            "sourcePath":
            "$.displayName",
            "targetPath": "$
            ['urn:sap:cloud:scim:schemas:exten
            sion:custom:2.0:Group']['name']",
            "functions": [

```

```

    {
      "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
      "targetPath":
"$ .displayName"
    },
    {
      "optional": true,

      "preserveArrayWithSingleElement":
true,
      "sourcePath": "$ .members",
      "targetPath": "$ .members"
    },
    {
      "constant":
"urn:sap:cloud:scim:schemas:extension:custom:2.0:Group",
      "targetPath":
"$ .schemas[1]"
    },
    {
      "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
      "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]"
    },
    {
      "optional": true,
      "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ]
[ 'description' ]",
      "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ]
[ 'description' ]"
    }
  ]
}

```

```

    {
      "type":
"replaceAllString",
      "regex": "[\\s\\
\\p{Punct}]",
      "replacement": "_"
    }
  ],
  {
    "scope": "createEntity",
    "optional": true,
    "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
    "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]"
  },
  {
    "optional": true,
    "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ]
[ 'description' ]",
    "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ]
[ 'description' ]"
  },
  {
    "optional": true,

    "preserveArrayWithSingleElement":
true,
    "sourcePath": "$ .members",
    "targetPath": "$ .members"
  }
]
}

```

### Default transformations for SCIM API version 2:

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$..meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['userUid']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUid']"
      },
      {
        "sourcePath": "$.schemas",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath":
"$..userName",
        "targetPath":
"$..userName",
        "optional": true,
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$..emails[*].value",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.emails[?
(@.value)]"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant":
["urn:ietf:params:scim:schemas:core:2.0:User", "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User", "urn:ietf:params:scim:schemas:extension:sap:2.0:User"],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath":
"$..userName",
        "targetPath":
"$..userName",
        "optional": true
      },
      {
        "sourcePath":
"$..emails[*].value",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.emails[?
(@.value)]"
      },
      {
        "sourcePath":
"$..userType",
        "targetPath":
"$..userType",
        "optional": true
      },
      {
        "sourcePath":
"$..name.givenName",
        "targetPath":
"$..name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$..name.middleName",
        "targetPath":
"$..name.middleName",
        "optional": true
      },
      {
        "sourcePath":
"$..name.familyName",
        "targetPath":
"$..name.familyName",

```

```

    },
    {
      "sourcePath":
"$$.emails[0].value",
      "targetPath":
"$$.emails[0].value"
    },
    {
      "sourcePath": "$$.emails[?
(@.primary== true)].value",
      "correlationAttribute":
true
    },
    {
      "sourcePath": "$$.active",
      "targetPath": "$$.active"
    },
    {
      "sourcePath":
"$$.userType",
      "targetPath":
"$$.userType",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.givenName",
      "targetPath":
"$$.name.givenName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.middleName",
      "targetPath":
"$$.name.middleName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.familyName",
      "targetPath":
"$$.name.familyName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "targetPath":
"$$.name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$$.addresses",
      "targetPath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },

```

```

      "optional": true
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "targetPath":
"$$.name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$$.addresses",
      "targetPath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "defaultValue": [],
      "optional": true,
      "functions": [
        {
          "function":
"putIfAbsent",
          "key": "type",
          "defaultValue": "work"
        },
        {
          "condition": "(@.type
NIN ['work', 'home'])",
          "function":
"putIfPresent",
          "key": "type",
          "defaultValue": "work"
        }
      ]
    },
    {
      "sourcePath": "$$.locale",
      "targetPath": "$$.locale",
      "optional": true
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "targetPath":
"$$.phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName",
      "optional": true
    },
    {
      "sourcePath": "$$
[urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User]"
      "validFrom": ""
    },

```

```

    {
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "optional": true
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "targetPath":
"$$.phoneNumbers",

      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.timezone",
      "targetPath":
"$$.timezone",
      "optional": true
    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName",
      "optional": true
    },
    {
      "sourcePath":
"$$.sourceSystem",
      "targetPath":
"$$.sourceSystem",
      "ignore": true
    },
    {
      "sourcePath": "$.groups",
      "targetPath": "$.groups",

      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "type": "remove",
      "targetPath":
"$$.groups[*].display"
    },
    {
      "condition":
"$$.displayName EMPTY true",
      "type": "remove",
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",

```

```

      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['validFrom']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['validTo']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['validTo']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext

```

```

        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "optional": true
    },
    {

```

```

        "targetPath": "$
['department']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "defaultValue": true,
        "optional": true
    },
    {
        "constant": false,
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['sendMail']",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$.emails",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']['emails']",
        "scope": "createEntity",
        "functions": [
            {
                "function":
"putIfAbsent",
                "key": "verified",
                "defaultValue": true
            }
        ]
    },
    {

```

```

      "sourcePath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['manager']['displayName'],
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['manager']['displayName'],
      "optional": true
    },
    {
      "sourcePath": "$
      ['urn:sap:cloud:scim:schemas:exten
      sion:custom:2.0:User'],
      "targetPath": "$
      ['urn:sap:cloud:scim:schemas:exten
      sion:custom:2.0:User'],
      "optional": true
    },
    {
      "sourcePath": "$.company",
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['organization'],
      "optional": true
    }
  ],
  "group": {
    "scimEntityEndpoint":
    "Groups",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
        "entityIdSourceSystem"
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [
          {
            "type":
            "concatString",
            "suffix": "$
            {entityIdSourceSystem}"
          }
        ]
      }
    ],
    {
      "sourcePath": "$
      ['urn:sap:cloud:scim:schemas:exten
      sion:custom:2.0:Group']['name'],
      "targetPath": "$
      ['urn:sap:cloud:scim:schemas:exten
      sion:custom:2.0:Group']['name']"
    }
  ]
}

```

```

      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']['emails']['*']
      ['type'],
      "type": "remove"
    },
    {
      "constant": "disabled",
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']
      ['passwordDetails']['status'],
      "scope": "createEntity"
    },
    {
      "constant": 39,
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']
      ['sourceSystem'],
      "scope": "createEntity"
    },
    {
      "constant": "employee",
      "targetPath": "$.userType"
    },
    {
      "sourcePath":
      "$.timezone",
      "targetPath":
      "$.timezone",
      "optional": true
    },
    {
      "sourcePath":
      "$.Operations",
      "targetPath":
      "$.Operations",
      "preserveArrayWithSingleElement":
      true,
      "scope": "patchEntity"
    },
    {
      "sourcePath": "$.schemas",
      "targetPath": "$.schemas",
      "preserveArrayWithSingleElement":
      true,
      "scope": "patchEntity"
    }
  ],
  "group": {
    "scimEntityEndpoint":
    "Groups",
    "mappings": [
      {
        "sourceVariable":
        "entityIdTargetSystem",
        "targetPath": "$.id"
      }
    ]
  }
}

```

```

    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath":
"$$.members",
      "targetPath":
"$$.members",

      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath": "$$.schemas",

      "preserveArrayWithSingleElement":
true,
      "targetPath": "$$.schemas"
    }
  ]
}

```

```

      "sourcePath":
"$$.Operations",
      "targetPath":
"$$.Operations",

      "preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    },
    {
      "sourcePath": "$$.schemas",
      "targetPath": "$$.schemas",

      "preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    },
    {
      "constant":
[ "urn:ietf:params:scim:schemas:core:2.0:Group", "urn:sap:cloud:scim:schemas:extension:custom:2.0:Group" ],
      "targetPath": "$$.schemas"
    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath": "$$.members",
      "targetPath": "$$.members",

      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
      "scope": "createEntity",
      "functions": [
        {
          "type":
"replaceAllString",
          "regex": "[\\s\\
\\p{Punct}]",
          "replacement": "_"
        }
      ]
    },
    {
      "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'name' ]",
      "optional": true,

```



```

        "targetPath": "$
    ['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name'],
        "scope": "createEntity"
    },
    {
        "sourcePath": "$
    ['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']
    ['description'],
        "optional": true,
        "targetPath": "$
    ['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']
    ['description']"
    }
  ]
}

```

### → Remember

If you set \$.userType to **"public"**, all passwords will be written by default in the Identity Authentication. Thus, all provisioned users will successfully sign in to Identity Authentication target system.

When \$.userType is set to **"employee"**, the sign-in behavior of the provisioned users depends on whether users have been created with or without a password, and where these passwords are stored. Thus, you need to modify the target transformations accordingly in order for the users to successfully sign in to the SAP Cloud Identity Services administration console.

To learn more, see [Guided Answers: Identity Authentication: Provisioned Users Can't Sign In](#) 

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

### **i** Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### **⚠** Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Identity Authentication: Documentation](#)

[Identity Authentication: SCIM REST API](#)

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.2 Local Identity Directory

Follow this procedure to set up Local Identity Directory as a proxy system.

### Prerequisites

#### i Note

The *Local Identity Directory* connector is available for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

### Context

Identity directory is the user store of SAP Cloud Identity Services. It provides a central place for storing and managing users, groups and custom schemas through the System for Cross-domain Identity Management 2.0 REST API, in short Identity Directory SCIM API.

You can use the identity directory for user and group provisioning with an external identity management system without making a direct connection between them. For this, you configure the Local Identity Directory connector as a proxy system in the SAP Cloud Identity Services administration console.

The entities are read and written to and from the identity directory of your current SAP Cloud Identity Services tenant. Therefore, you don't have to set up connectivity and authentication properties for the proxy system. In case you want to use the identity directory in another tenant, add Identity Authentication (version 2) as a proxy system with the respective connectivity set up.

The proxy systems consume [Identity Directory SCIM API](#). It supports patch operations for proxy systems only, paging for group members and user's groups, custom attributes, delta read mode for users. Also, the group resource mapping in the transformation is not ignored by default.

By configuring the directory as a proxy system, you implement secure authentication, single sign-on (SSO), strong authentication, and mobile SSO so that the provisioned users to the directory have access to the business applications of your company.

To create Local Identity Directory as a proxy system, proceed as follows:

### Procedure

1. Sign in to SAP Cloud Identity Services administration console and create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to SAP Cloud Identity Services.

If you already have a technical user, skip this step.

- For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*
  - For **Basic authentication**, navigate to ► [Users & Authorizations](#) ► [Administrators](#) and add an admin user of type **System**. Configure a client ID and secret for this user and enable the [Access Proxy System API](#) permission.
2. Navigate to ► [Identity Provisioning](#) ► [Proxy Systems](#) ►
  3. Add [Local Identity Directory](#) as a proxy system.

For more information, see [Add a System \[page 1477\]](#).

4. **Optional:** Configure properties. For example, properties to enable paging:  
`ids.user.groups.paging.enabled`.

Local Identity Directory properties are prefixed with `ids.<property_name>`. For more information, see [List of Properties \[page 94\]](#).

5. **Optional:** Configure transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the Local Identity Directory proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your Local Identity Directory. For more information, see:

- [Manage Transformations \[page 1494\]](#)
- SCIM API version 2: [Identity Directory SCIM API](#) 📄

Default read and write transformations:

#### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy [Read Transformation](#) is used for *write* cases, as well.

**Default transformations for Local Identity Directory:**

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$..meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['userUid']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['userUid']"
      },
      {
        "sourcePath": "$.schemas",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath":
"$..userName",
        "targetPath":
"$..userName",
        "optional": true,
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$..emails[*].value",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.emails[?
(@.value)]"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant":
["urn:ietf:params:scim:schemas:core:2.0:User", "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User", "urn:ietf:params:scim:schemas:extension:sap:2.0:User"],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath":
"$..userName",
        "targetPath":
"$..userName",
        "optional": true
      },
      {
        "sourcePath":
"$..emails[*].value",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.emails[?
(@.value)]"
      },
      {
        "sourcePath":
"$..userType",
        "targetPath":
"$..userType",
        "optional": true
      },
      {
        "sourcePath":
"$..name.givenName",
        "targetPath":
"$..name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$..name.middleName",
        "targetPath":
"$..name.middleName",
        "optional": true
      },
      {
        "sourcePath":
"$..name.familyName",
        "targetPath":
"$..name.familyName",

```

```

    },
    {
      "sourcePath":
"$$.emails[0].value",
      "targetPath":
"$$.emails[0].value"
    },
    {
      "sourcePath": "$$.emails[?
(@.primary== true)].value",
      "correlationAttribute":
true
    },
    {
      "sourcePath": "$$.active",
      "targetPath": "$$.active"
    },
    {
      "sourcePath":
"$$.userType",
      "targetPath":
"$$.userType",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.givenName",
      "targetPath":
"$$.name.givenName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.middleName",
      "targetPath":
"$$.name.middleName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.familyName",
      "targetPath":
"$$.name.familyName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "targetPath":
"$$.name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$$.addresses",
      "targetPath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },

```

```

      "optional": true
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "targetPath":
"$$.name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$$.addresses",
      "targetPath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "defaultValue": [],
      "optional": true,
      "functions": [
        {
          "function":
"putIfAbsent",
          "key": "type",
          "defaultValue": "work"
        },
        {
          "condition": "(@.type
NIN ['work', 'home'])",
          "function":
"putIfPresent",
          "key": "type",
          "defaultValue": "work"
        }
      ]
    },
    {
      "sourcePath": "$$.locale",
      "targetPath": "$$.locale",
      "optional": true
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "targetPath":
"$$.phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName",
      "optional": true
    },
    {
      "sourcePath": "$$
[urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User]"
      "validFrom": ""
    },

```

```

    {
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "optional": true
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "targetPath":
"$$.phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.timezone",
      "targetPath":
"$$.timezone",
      "optional": true
    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName",
      "optional": true
    },
    {
      "sourcePath":
"$$.sourceSystem",
      "targetPath":
"$$.sourceSystem",
      "ignore": true
    },
    {
      "sourcePath": "$.groups",
      "targetPath": "$.groups",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "type": "remove",
      "targetPath":
"$$.groups[*].display"
    },
    {
      "condition":
"$$.displayName EMPTY true",
      "type": "remove",
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",

```

```

      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['validFrom']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['validTo']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['validTo']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext

```

```

        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "optional": true
    },
    {

```

```

        "targetPath": "$
['department']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "defaultValue": true,
        "optional": true
    },
    {
        "constant": false,
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['sendMail']",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$.emails",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']['emails']",
        "scope": "createEntity",
        "functions": [
            {
                "function":
"putIfAbsent",
                "key": "verified",
                "defaultValue": true
            }
        ]
    },
    {

```



```

      "sourcePath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['manager']['displayName'],
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['manager']['displayName'],
      "optional": true
    },
    {
      "sourcePath": "$
      ['urn:sap:cloud:scim:schemas:exten
      sion:custom:2.0:User'],
      "targetPath": "$
      ['urn:sap:cloud:scim:schemas:exten
      sion:custom:2.0:User'],
      "optional": true
    },
    {
      "sourcePath": "$.company",
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['organization'],
      "optional": true
    }
  ],
  "group": {
    "scimEntityEndpoint":
    "Groups",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
        "entityIdSourceSystem"
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [
          {
            "type":
            "concatString",
            "suffix": "$
            {entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath": "$
        ['urn:sap:cloud:scim:schemas:exten
        sion:custom:2.0:Group']['name'],
        "targetPath": "$
        ['urn:sap:cloud:scim:schemas:exten
        sion:custom:2.0:Group']['name']"
      }
    ]
  }
}

```

```

      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']['emails']['*']
      ['type'],
      "type": "remove"
    },
    {
      "constant": "disabled",
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']
      ['passwordDetails']['status'],
      "scope": "createEntity"
    },
    {
      "constant": 39,
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']
      ['sourceSystem'],
      "scope": "createEntity"
    },
    {
      "constant": "employee",
      "targetPath": "$.userType"
    },
    {
      "sourcePath":
      "$.timezone",
      "targetPath":
      "$.timezone",
      "optional": true
    },
    {
      "constant": "userName",
      "targetVariable":
      "entityCorrelationAttributeName"
    },
    {
      "sourcePath":
      "$.userName",
      "targetVariable":
      "entityCorrelationAttributeValue"
    },
    {
      "sourcePath":
      "$.Operations",
      "targetPath":
      "$.Operations",
      "preserveArrayWithSingleElement":
      true,
      "scope": "patchEntity"
    },
    {
      "sourcePath": "$.schemas",
      "targetPath": "$.schemas",
      "preserveArrayWithSingleElement":
      true,
      "scope": "patchEntity"
    }
  ]
}

```

```

    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath":
"$$.members",
      "targetPath":
"$$.members",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath": "$$.schemas",

      "preserveArrayWithSingleElement":
true,
      "targetPath": "$$.schemas"
    }
  ]
}

```

```

    },
    "group": {
      "scimEntityEndpoint":
"Groups",
      "mappings": [
        {
          "sourceVariable":
"entityIdTargetSystem",
          "targetPath": "$.id"
        },
        {
          "sourcePath":
"$$.Operations",
          "targetPath":
"$$.Operations",
          "preserveArrayWithSingleElement":
true,
          "scope": "patchEntity"
        },
        {
          "sourcePath": "$$.schemas",
          "targetPath": "$$.schemas",
          "preserveArrayWithSingleElement":
true,
          "scope": "patchEntity"
        },
        {
          "constant":
["urn:ietf:params:scim:schemas:core:2.0:Group", "urn:sap:cloud:scim:schemas:extension:custom:2.0:Group"],
          "targetPath": "$$.schemas"
        },
        {
          "sourcePath":
"$$.displayName",
          "targetPath":
"$$.displayName"
        },
        {
          "sourcePath": "$$.members",
          "targetPath": "$$.members",
          "preserveArrayWithSingleElement":
true,
          "optional": true
        },
        {
          "sourcePath":
"$$.displayName",
          "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
          "scope": "createEntity",
          "functions": [
            {
              "type":
"replaceAllString",

```

```

        "regex": "[\\s\\
\\p{Punct}]",
        "replacement": "_"
    }
  ],
  {
    "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
    "optional": true,
    "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
    "scope": "createEntity"
  },
  {
    "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']
['description']",
    "optional": true,
    "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']
['description']"
  }
]
}
}

```

### → Remember

If you set \$.userType to **"public"**, all passwords will be written by default in the Local Identity Directory. Thus, all provisioned users will successfully sign in to Local Identity Directory target system.

When \$.userType is set to **"employee"**, the sign-in behavior of the provisioned users depends on whether users have been created with or without a password, and where these passwords are stored. Thus, you need to modify the target transformations accordingly in order for the users to successfully sign in to the SAP Cloud Identity Services administration console.

To learn more, see [Guided Answers: Identity Authentication: Provisioned Users Can't Sign In](#) 

6. Connect the external consumer to SAP Cloud Identity Services with the technical user you have created in step 1.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### i Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

## Related Information

[Identity Directory \[page 1567\]](#)

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.3 SAP Advanced Financial Closing

Follow this procedure to set up SAP Advanced Financial Closing as a proxy system.

### Prerequisites

- You have created an instance and generated a service key for the standard service plan of SAP Advanced Financial Closing. For more information, see: [How to Manage User Access Using the SCIM API Provided](#). The service key contains the API URL and the OAuth credentials (`clientId` and `clientsecret`) under the `uaa` property.

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

SAP Advanced Financial Closing allows you to define, automate, process, and monitor the financial closing tasks for the entities of your organization. It is an SAP BTP application that runs in an SAP BTP subaccount.

You can use Identity Provisioning to configure SAP Advanced Financial Closing as a proxy system and configure it in hybrid scenarios. For example, when SAP Advanced Financial Closing is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management and SAP Cloud Identity Access Governance, without making a direct connection between both systems. You can provision users, user groups and user roles to the external backend system, which can trigger CRUD (create, read, update, delete) operations on users, user groups and user roles back to the SAP Advanced Financial Closing.

## Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit ► [Neo](#) ► [Overview](#) ►, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Advanced Financial Closing](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note





If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>

Property Name	Value
URL	<p>Enter the URL provided by the service key under  <i>endpoints</i>  <i>scim2</i>  without adding the path information.</p> <p>For example: <code>https://afc-production-afc-api.cfapps.eu10.hana.ondemand.com</code></p>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user ID provided by the service key under  <i>uaa</i>  <i>clientid</i>  .
Password	(Credential) Enter the password provided by the service key under  <i>uaa</i>  <i>clientsecret</i>  .
OAuth2TokenServiceURL	<p>Enter the OAuth 2.0 Token Service URL provided by the service key of your SAP Advanced Financial Closing instance. It follows the pattern: <i>&lt;uaa.url&gt;/oauth/token</i>, where:</p> <ul style="list-style-type: none"> <li><i>&lt;uaa.url&gt;</i> is the URL provided by the service key under  <i>uaa</i>  <i>url</i> .</li> <li><i>/oauth/token</i> is the suffix you need to add.</li> </ul>
(Optional) <code>s4hana.afc.user.filter</code>	<p>When specified, only those SAP Advanced Financial Closing users matching the filter expression will be read.</p> <p>Supported operators: <b>eq</b> (equal), <b>sw</b> (starts with) and <b>co</b> (contains)</p> <p>For example:</p> <ul style="list-style-type: none"> <li><i>userName eq "Julie Armstrong"</i></li> <li><i>emails eq "julie.armstrong@example.com"</i></li> <li><i>name.familyName sw "A"</i></li> <li><i>name.givenName co "Ju"</i></li> </ul>
(Optional) <code>s4hana.afc.group.filter</code>	<p>When specified, only those SAP Advanced Financial Closing users matching the filter expression will be read.</p> <p>Supported operators: <b>eq</b> (equal), <b>sw</b> (starts with) and <b>co</b> (contains)</p> <p>For example: <i>displayName eq "Administrators"</i></p>

Property Name	Value
<code>s4hana.afc.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <code>userName</code></li> <li>• <code>emails[0].value</code></li> <li>• <code>userName,emails[0].value</code></li> </ul>
<code>s4hana.afc.group.unique.attribute</code>	<p>If Identity Provisioning tries to provision a group that already exists in the target system (a conflicting group), this property defines the unique attributes by which the existing group will be searched and resolved.</p> <p>The default value is <code>displayName</code>.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map user and group attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Advanced Financial Closing* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Advanced Financial Closing system. For more information, see: [Manage Transformations \[page 1494\]](#).

[How to Manage User Access Using the SCIM API Provided](#)

[SAP Business Accelerator Hub: SAP Advanced Financial Closing](#) 

Default read and write transformations:



## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath":
        "$.schemas",
        "targetPath":
        "$.schemas",
        "preserveArrayWithSingleElement":
        true,
        "optional": true
      },
      {
        "sourcePath": "$
        ['urn:ietf:params:scim:schemas:ext
        ension:sap:2.0:User']
        ['userUuid']",
        "targetPath": "$
        ['urn:ietf:params:scim:schemas:ext
        ension:sap:2.0:User']
        ['userUuid']",
        "optional": true
      },
      {
        "sourcePath":
        "$.id",
        "targetPath":
        "$.id",
        "targetVariable":
        "entityIdSourceSystem"
      },
      {
        "sourcePath":
        "$.userName",
        "targetPath":
        "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
        "$.name.formatted",
        "targetPath":
        "$.name.formatted",
        "optional": true
      },
      {
        "sourcePath":
        "$.name.familyName",
        "targetPath":
        "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath":
        "$.name.givenName",

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "constant": [
        "urn:ietf:params:scim:schemas:core
        :2.0:User",
        "urn:ietf:params:scim:schemas:exte
        nsion:enterprise:2.0:User",
        "urn:ietf:params:scim:schemas:exte
        nsion:sap:2.0:User"
        ],
        "targetPath":
        "$.schemas"
      },
      {
        "sourcePath": "$
        ['urn:ietf:params:scim:schemas:ext
        ension:sap:2.0:User']
        ['userUuid']",
        "targetPath": "$
        ['urn:ietf:params:scim:schemas:ext
        ension:sap:2.0:User']
        ['userUuid']",
        "optional": true
      },
      {
        "sourceVariable":
        "entityIdTargetSystem",
        "targetPath":
        "$.id"
      },
      {
        "sourcePath":
        "$.userName",
        "targetPath":
        "$.userName"
      },
      {
        "sourcePath":
        "$.name.formatted",
        "targetPath":
        "$.name.formatted",
        "optional": true
      },
      {
        "sourcePath":
        "$.name.familyName",
        "targetPath":
        "$.name.familyName",
        "optional": true
      },
      {
        "sourcePath":
        "$.name.givenName",

```

```

        "targetPath":
"$ .name.givenName",
        "optional": true
    },
    {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName",
        "optional": true
    },
    {
        "sourcePath":
"$ .externalId",
        "targetPath":
"$ .externalId",
        "optional": true
    },
    {
        "sourcePath":
"$ .locale",
        "targetPath":
"$ .locale",
        "optional": true
    },
    {
        "sourcePath":
"$ .active",
        "targetPath":
"$ .active",
        "optional": true
    },
    {
        "sourcePath":
"$ .emails",
        "targetPath":
"$ .emails",
        "preserveArrayWithSingleElement":
true,
        "optional": true
    },
    {
        "sourcePath":
"$ .groups",
        "targetPath":
"$ .groups",
        "preserveArrayWithSingleElement":
true,
        "optional": true
    },
    {
        "sourcePath":
"$ .roles",
        "targetPath":
"$ .roles",
        "preserveArrayWithSingleElement":
true,
        "optional": true
    },

```

```

        "targetPath":
"$ .name.givenName",
        "optional": true
    },
    {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName",
        "optional": true
    },
    {
        "sourcePath":
"$ .locale",
        "targetPath":
"$ .locale",
        "optional": true
    },
    {
        "sourcePath":
"$ .active",
        "targetPath":
"$ .active",
        "optional": true
    },
    {
        "condition":
"$ .emails[0].length() > 0",
        "targetPath":
"$ .emails[0].primary",
        "constant": true
    },
    {
        "sourcePath":
"$ .emails",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$ .emails",
        "functions": [
            {
                "function": "putIfAbsent",
                "key":
"type",
                "defaultValue": "work"
            }
        ],
        "sourcePath":
"$ .phoneNumbers",
        "targetPath":
"$ .phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "optional": true
    },

```

```

    {
      "sourcePath":
"$ .phoneNumbers",
      "targetPath":
"$ .phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$ .meta",
      "targetPath":
"$ .meta",
      "optional": true
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetPath":
"$ .meta.location",
      "targetVariable":
"entityLocationSourceSystem",
      "functions": [
        {
          "type":
"concatString",
          "suffix":
"${entityIdSourceSystem}"
        }
      ]
    }
  ],
  "group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
      {
        "sourcePath":
"$ .id",
        "targetPath":
"$ .id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName"
      },
      {
        "sourcePath":
"$ .members",
        "targetPath":
"$ .members",
        "preserveArrayWithSingleElement":
true,
        "optional": true
      }
    ]
  }
}

```

```

    "sourcePath":
"$ .roles",
    "targetPath":
"$ .roles",
    "preserveArrayWithSingleElement":
true,
    "optional": true
  },
  {
    "constant":
"urn:ietf:params:scim:api:messages:2.0:PatchOp",
    "targetPath":
"$ .schemas[0]",
    "scope":
"patchEntity"
  },
  {
    "sourcePath":
"$ .Operations",
    "preserveArrayWithSingleElement":
true,
    "targetPath":
"$ .Operations",
    "scope":
"patchEntity"
  }
],
"group": {
  "scimEntityEndpoint":
"Groups",
  "mappings": [
    {
      "sourceVariable":
"entityIdTargetSystem",
      "targetPath":
"$ .id"
    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath":
"$ .schemas[0]"
    },
    {
      "sourcePath":
"$ .displayName",
      "targetPath":
"$ .displayName"
    },
    {
      "sourcePath":
"$ .members",
      "targetPath":
"$ .members",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    }
  ]
}

```

## Read Transformation

```

    },
    {
      "sourcePath":
"$$.schemas",
      "targetPath":
"$$.schemas",
      "preserveArrayWithSingleElement":
true
    },
    {
      "sourcePath":
"$$.meta",
      "targetPath":
"$$.meta",
      "optional": true
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetPath":
"$$.meta.location",
      "targetVariable":
"entityLocationSourceSystem",
      "functions": [
        {
          "type":
"concatString",
          "suffix":
"${entityIdSourceSystem}"
        }
      ]
    }
  ]
}

```

## Write Transformation

```

    },
    {
      "sourcePath":
"$$.members[*].value",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$$.members[?(@.value)]",
      "functions": [
        {
          "function": "resolveEntityIds"
        }
      ]
    },
    {
      "sourcePath":
"$$.Operations",
      "targetPath":
"$$.Operations",
      "preserveArrayWithSingleElement":
true,
      "scope":
"patchEntity"
    },
    {
      "sourcePath":
"$$.schemas",
      "targetPath":
"$$.schemas",
      "preserveArrayWithSingleElement":
true,
      "scope":
"patchEntity"
    }
  ]
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.4 SAP Advanced Workflow

Follow this procedure to set up SAP Advanced Workflow as a proxy system.

### Prerequisites

- You have technical credentials for SAP Advanced Workflow. Note that SAP Advanced Workflow is available to SAP Commissions customers as an optional add-on. You create an Admin user in SAP Commissions, which is synchronized with SAP Advanced Workflow. For more information, see: [Adding an Admin User](#) and [Commissions User Synchronization](#).
- You have set up SSO between Identity Authentication and SAP Advanced Workflow. For more information, see [Integration with SAP IdP](#).

### Context


SAP Advanced Workflow enables you to analyse, organize, and execute business processes to connect people, data, and daily activities. Workflow provides you the tools you need to configure and customize your business processes based on your specific business needs.

Create an SAP Advanced Workflow proxy connector to execute hybrid scenarios. That means, it can provision its users to another (external) back-end system by request, and then can continue executing CRUD operations back to the SAP Advanced Workflow system, whenever the external back-end requests such. This scenario supports provisioning **users**.

#### i Note

SAP Advanced Workflow does not support groups.

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example:  
`GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'`
- If your system supports multivalued e-mails (that is `$.emails[0].value`, `$.emails[1].value`, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (`$.emails[0].value`).

### ❁ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `awf.user.filter = timezone eq "Africa"`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=username eq "johnsmith03"`

The query request to the SAP Central Business Configuration API will result into: `/Users?filter=timezone eq "Africa" and username eq "johnsmith03"`

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Advanced Workflow* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.



## Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your SAP Advanced Workflow system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the user for your SAP Advanced Workflow system.
Password	(Credential) Enter the password for your SAP Advanced Workflow user.
awf.domain	<p>The domain name is the name of your SAP Advanced Workflow tenant.</p> <p>If you don't know your tenant name, contact your supervisor or administrator, or refer to the email notification you received when your account was created.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Advanced Workflow](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Advanced Workflow. For more information, see:

[Manage Transformations \[page 1494\]](#)

Default read and write transformations:

### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints ([/Users](#) or [/Groups](#)) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy [Read Transformation](#) is used for [write](#) cases, as well.

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourcePath":
"$$.schemas",
        "targetPath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.name.givenName",
        "optional": true,
        "targetPath":
"$$.name.givenName"
      },
      {
        "sourcePath":
"$$.name.middleName",
        "optional": true,
        "targetPath":
"$$.name.middleName"
      },
      {
        "sourcePath":
"$$.name.familyName",
        "optional": true,
        "targetPath":
"$$.name.familyName"
      },
      {
        "sourcePath":
"$$.phoneNumbers[0].value",
        "optional": true,
        "targetPath":
"$$.phoneNumbers[0].value"
      },
      {
        "sourcePath":
"$$.profileUrl",

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "constant": [
"urn:ietf:params:scim:schemas:core:2.0:User",
"urn:ietf:params:scim:schemas:extension:sap.spm.workflow:2.0:User"
        ],
        "targetPath":
"$$.schemas"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName"
      },
      {
        "sourcePath":
"$$.name.givenName",
        "optional": true,
        "targetPath":
"$$.name.givenName"
      },
      {
        "sourcePath":
"$$.name.middleName",
        "optional": true,
        "targetPath":
"$$.name.middleName"
      },
      {
        "sourcePath":
"$$.name.familyName",
        "optional": true,
        "targetPath":
"$$.name.familyName"
      },
      {
        "sourcePath":
"$$.phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.phoneNumbers"
      },

```

```

      "optional": true,
      "targetPath":
"$ .profileUrl"
    },
    {
      "sourcePath":
"$ .active",
      "optional": true,
      "targetPath":
"$ .active"
    },
    {
      "sourcePath":
"$ .emails[0].value",
      "optional": true,
      "targetPath":
"$ .emails[0].value"
    },
    {
      "sourcePath":
"$ .timezone",
      "optional": true,
      "targetPath":
"$ .timezone"
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetPath":
"$ .meta.location",
      "targetVariable":
"entityLocationSourceSystem",
      "functions": [
        {
          "type":
"concatString",
          "suffix":
"${entityIdSourceSystem}"
        }
      ]
    },
    {
      "scimEntityEndpoint":
"Users"
    }
  }
}

```

```

      "sourcePath":
"$ .active",
      "optional": true,
      "targetPath":
"$ .active"
    },
    {
      "sourcePath":
"$ .emails",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$ .emails"
    },
    {
      "sourcePath":
"$ .timezone",
      "optional": true,
      "targetPath":
"$ .timezone"
    },
    {
      "constant":
"urn:ietf:params:scim:api:messages:2.0:PatchOp",
      "targetPath":
"$ .schemas[0]",
      "scope":
"patchEntity"
    },
    {
      "sourcePath":
"$ .Operations",
      "preserveArrayWithSingleElement":
true,
      "targetPath":
"$ .Operations",
      "scope":
"patchEntity"
    },
    {
      "scimEntityEndpoint":
"Users"
    }
  }
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## 1.6.3.5 SAP Analytics Cloud

Follow this procedure to set up SAP Analytics Cloud as a proxy system.

### Prerequisites

- In SAP Analytics Cloud, you have enabled a custom SAML Identity Provider, for which *User Attribute* is set to **Custom SAML User Mapping**. To learn how, see: [Enabling a Custom SAML Identity Provider](#)
- In SAP Analytics Cloud, you have added an OAuth client with authorization grant **Client Credentials**. To learn how, see: [Managing OAuth Clients and Trusted Identity Providers](#)

#### i Note

Administrators of bundle tenants on Neo environment should enable the *Manage OAuth Clients* permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

SAP Analytics Cloud is an all-in-one cloud product offered as software as a service for business intelligence, planning, and predictive analytics.

You can use Identity Provisioning to configure SAP Analytics Cloud as a proxy system in hybrid scenarios. For example, when SAP Analytics Cloud is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management, without making a direct connection between both systems. You can provision users and groups to the external backend system, which can trigger CRUD (create, read, update, delete) operations on users and groups back to the SAP Analytics Cloud.


There are two versions of the SAP Analytics Cloud SCIM API. They are handled by the `sac.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on April 10, 2023), SAP Analytics Cloud SCIM API version 1 is used. This is the default value.
- When the value is set to **2** - SAP Analytics Cloud SCIM API version 2 is used. This version is released with enhancements, among which the support for patch operations.

For more information on the differences between SAP Analytics Cloud SCIM API version 1 and 2, see [Managing Users and Teams](#).

For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#).

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports *native read filtering*, the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the *Read Transformation*, there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
*GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the *Properties* tab: *scim.user.filter = timezone eq "US"*

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP Analytics Cloud API will result into: **/Users?filter=timezone eq "US" and userName eq "johnsmith03"**

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

#### i Note

If you have a bundle tenant, then in the cockpit → *Neo* → *Overview*, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in

the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li> <li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li> <li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li> <li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Analytics Cloud* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter the URL to your SAP Analytics Cloud system without adding the path information.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the client ID to retrieve the OAuth access token for SAP Analytics Cloud.
Password	(Credential) Enter the client secret to retrieve the OAuth access token for SAP Analytics Cloud.
OAuth2TokenServiceURL	<p>Enter the URL of the access token provider service for your SAP Analytics Cloud instance.</p> <p>This token URL is listed in the <a href="#">OAuth Clients</a> section of the <a href="#">App Integration</a> page. For more information, refer to <a href="#">Authorize API Access for OAuth Clients</a> in <a href="#">Manage OAuth Clients</a></p>
scim.api.csrf.protection	<p>Specifies whether to fetch a CSRF token when sending requests to the system.</p> <p>This property is automatically added to the system, with default value: <b>enabled</b></p>
csrf.token.path	<p>Path which is appended to the URL to retrieve the CSRF token.</p> <p>This property is automatically added in the system, with default value: <b>/api/v1/scim/Users?count=1</b></p>
(Optional) sac.api.version	<p>Handles the version of SAP Analytics Cloud SCIM API.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><a href="#">1</a> - Indicates that SAP Analytics Cloud SCIM API version 1 is used.</li> <li><a href="#">2</a> - Indicates that SAP Analytics Cloud SCIM API version 2 is used.</li> </ul> <p>Default value: <a href="#">1</a></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 6. Configure the transformations.



Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Analytics Cloud](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Analytic Cloud. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Analytics Cloud: User and Team Provisioning API](#)

[Managing Users and Teams → api/v1/scim](#)

[Managing Users and Teams → api/v1/scim2](#)

Default read and write transformations:

#### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy [Read Transformation](#) is used for [write](#) cases, as well.

## Default read and write transformations for SAP Analytics Cloud SCIM API version 1:

### Read Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",

"preserveArrayWithSingleElement":
true
      },
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta .location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name"
      },
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",
```

### Write Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "condition":
"($.emails[0].value EMPTY false)",
    "mappings": [
      {
        "sourcePath": "$.schemas",

"preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$.schemas"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath":
"$ .emails [0] .value",
        "targetPath": "$.userName"
      },
      {
        "condition": "$.emails[?
(@.primary == true)].value != []",
        "sourcePath": "$.emails[?
(@.primary == true)].value",

"preserveArrayWithSingleElement":
false,
        "optional": true,
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "optional": true,
        "targetPath": "$.name"
      },
      {
        "sourcePath":
"$ .displayName",
        "optional": true,
        "targetPath":
"$ .displayName"
      },
      {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails",

"preserveArrayWithSingleElement":
true,
```

```

        "targetPath": "$.emails",

"preserveArrayWithSingleElement":
true
    },
    {
        "sourcePath":
        "$.emails[0].value",
        "targetPath":
        "$.emails[0].value"
    },
    {
        "sourcePath": "$.emails[?
(@.primary== true)].value",
        "correlationAttribute":
true
    },
    {
        "sourcePath": "$.roles",
        "targetPath": "$.roles",

"preserveArrayWithSingleElement":
true
    },
    {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",

"preserveArrayWithSingleElement":
true
    },
    {
        "sourcePath": "$
['urn:scim:schemas:extension:enter
prise:1.0']['manager']
['managerId']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']"
    }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "sourcePath": "$.id",
                "targetPath": "$.id",
                "targetVariable":
"entityIdSourceSystem"
            },
            {
                "sourceVariable":
"entityBaseLocation",
                "targetVariable":
"entityLocationSourceSystem",
                "targetPath":
"$$.meta.location",
                "functions": [
                    {

```

```

        "targetPath": "$.emails"
    },
    {
        "sourcePath": "$.roles",

"preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$.roles"
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
        "optional": true,
        "targetPath": "$
['urn:scim:schemas:extension:enter
prise:1.0']['manager']
['managerId']"
    }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "sourcePath": "$.schemas",

"preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$.schemas"
    },
    {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
    },
    {
        "sourcePath":
"$$.displayName",
        "targetPath": "$.id",
        "scope": "createEntity"
    },
    {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
    },
    {
        "sourcePath": "$.roles",

"preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$.roles"
    },
    {
        "sourcePath": "$.members",

```

## Read Transformation

```

      "type":
"concatString",
      "suffix": "$
{entityIdSourceSystem}"
    ]
  },
  {
    "sourcePath":
"$$.displayName",
    "targetPath":
"$$.displayName"
  },
  {
    "sourcePath": "$.members",
    "targetPath": "$.members",

"preserveArrayWithSingleElement":
true
  },
  {
    "sourcePath": "$.schemas",
    "targetPath": "$.schemas",

"preserveArrayWithSingleElement":
true
  },
  {
    "sourcePath": "$.roles",
    "targetPath": "$.roles",

"preserveArrayWithSingleElement":
true
  }
]
}
}

```

## Write Transformation

```

"preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath": "$.members"
    ]
  }
}

```

Default read and write transformations for SAP Analytics Cloud SCIM API version 2:

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",

"preserveArrayWithSingleElement":
true
      },
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta .location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ],
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath":
"$ .externalId",
        "targetPath":
"$ .externalId",
        "optional": true
      },
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName",
        "optional": true
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "condition":
"($.emails[0].value EMPTY false)",
    "mappings": [
      {
        "constant": [

"urn:sap:params:scim:schemas:exten
sion:sac:2.0:user-custom-
parameters",

"urn:ietf:params:scim:schemas:core
:2.0:User",

"urn:ietf:params:scim:schemas:exte
nsion:enterprise:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath":
"$ .emails[0].value",
        "targetPath": "$.userName"
      },
      {
        "condition": "$.emails[?
(@.primary == true)].value != []",
        "sourcePath": "$.emails[?
(@.primary == true)].value",

"preserveArrayWithSingleElement":
false,
        "optional": true,
        "targetPath": "$.userName"
      },
      {
        "sourcePath":
"$ .userName",
        "optional": true,
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "optional": true,
        "targetPath": "$.name"
      },
      {
        "sourcePath":
"$ .displayName",
        "optional": true,
        "targetPath":
"$ .displayName"
      }
    ]
  }
}

```

```

    },
    {
      "sourcePath": "$.active",
      "targetPath": "$.active"
    },
    {
      "sourcePath": "$.emails",
      "targetPath": "$.emails",
    },
    "preserveArrayWithSingleElement":
    true
  },
  {
    "sourcePath":
    "$.emails[0].value",
    "targetPath":
    "$.emails[0].value"
  },
  {
    "sourcePath": "$.emails[?
    (@.primary== true)].value",
    "correlationAttribute":
    true
  },
  {
    "sourcePath": "$.roles",
    "targetPath": "$.roles",
  },
  "preserveArrayWithSingleElement":
  true,
  "optional": true
},
{
  "sourcePath": "$.groups",
  "targetPath": "$.groups",
},
"preserveArrayWithSingleElement":
true,
"optional": true
},
{
  "sourcePath": "$
  ['urn:sap:params:scim:schemas:exte
  nsion:sac:2.0:user-custom-
  parameters']",
  "targetPath": "$
  ['urn:sap:params:scim:schemas:exte
  nsion:sac:2.0:user-custom-
  parameters']"
},
{
  "sourcePath": "$
  ['urn:ietf:params:scim:schemas:ext
  ension:enterprise:2.0:User']",
  "targetPath": "$
  ['urn:ietf:params:scim:schemas:ext
  ension:enterprise:2.0:User']",
  "optional": true
}
],
},
"group": {

```

```

    {
      "sourcePath":
      "$.externalId",
      "optional": true,
      "targetPath":
      "$.externalId"
    },
    {
      "sourcePath": "$.active",
      "optional": true,
      "targetPath": "$.active"
    },
    {
      "sourcePath": "$.emails",
    },
    "preserveArrayWithSingleElement":
    true,
    "targetPath": "$.emails"
  },
  {
    "condition":
    "$.emails[0].length() > 0",
    "constant": true,
    "targetPath":
    "$.emails[0].primary"
  },
  {
    "sourcePath": "$
    ['urn:sap:params:scim:schemas:exte
    nsion:sac:2.0:user-custom-
    parameters']",
    "optional": true,
    "targetPath": "$
    ['urn:sap:params:scim:schemas:exte
    nsion:sac:2.0:user-custom-
    parameters']"
  },
  {
    "sourcePath":
    "$.emails[0].value",
    "optional": true,
    "targetPath": "$
    ['urn:sap:params:scim:schemas:exte
    nsion:sac:2.0:user-custom-
    parameters']['idpUserId']"
  },
  {
    "condition": "$.emails[?
    (@.primary == true)].value != []",
    "sourcePath": "$.emails[?
    (@.primary == true)].value",
  },
  "preserveArrayWithSingleElement":
  false,
  "optional": true,
  "targetPath": "$
  ['urn:sap:params:scim:schemas:exte
  nsion:sac:2.0:user-custom-
  parameters']['idpUserId']"
},
{
  "sourcePath": "$
  ['urn:sap:params:scim:schemas:exte

```

```

    "scimEntityEndpoint":
"Groups",
    "mappings": [
        {
            "sourcePath": "$.id",
            "targetPath": "$.id",
            "targetVariable":
"entityIdSourceSystem"
        },
        {
            "sourceVariable":
"entityBaseLocation",
            "targetVariable":
"entityLocationSourceSystem",
            "targetPath":
"$ .meta.location",
            "functions": [
                {
                    "type":
"concatString",
                    "suffix": "$
{entityIdSourceSystem}"
                }
            ],
            "sourcePath":
"$ .displayName",
            "targetPath":
"$ .displayName"
        },
        {
            "sourcePath": "$.members",
            "targetPath": "$.members",
            "optional": true,

"preserveArrayWithSingleElement":
true
        },
        {
            "sourcePath": "$.schemas",
            "targetPath": "$.schemas",

"preserveArrayWithSingleElement":
true
        },
        {
            "sourcePath": "$
['urn:sap:params:scim:schemas:exten
sion:sac:2.0:group-roles']",
            "targetPath": "$
['urn:sap:params:scim:schemas:exten
sion:sac:2.0:group-roles']",
            "optional": true
        },
        {
            "sourcePath": "$
['urn:sap:params:scim:schemas:exten
sion:sac:2.0:group-custom-
parameters']",
            "targetPath": "$
['urn:sap:params:scim:schemas:exte

```

```

nsion:sac:2.0:user-custom-
parameters']['idpUserId']",
            "optional": true,
            "targetPath": "$
['urn:sap:params:scim:schemas:exte
nsion:sac:2.0:user-custom-
parameters']['idpUserId']"
        },
        {
            "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']",
            "optional": true,
            "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']"
        },
        {
            "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
            "optional": true,
            "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']"
        }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "condition":
"('%sac.group.prefix%' ==
'null') || ($.displayName =~ /
%sac.group.prefix%.*)/",
        "mappings": [
            {
                "constant": [

"urn:ietf:params:scim:schemas:core
:2.0:Group",

"urn:sap:params:scim:schemas:exten
sion:sac:2.0:group-roles",

"urn:sap:params:scim:schemas:exten
sion:sac:2.0:group-custom-
parameters"
                ],
                "targetPath": "$.schemas"
            },
            {
                "sourceVariable":
"entityIdTargetSystem",
                "targetPath": "$.id"
            },
            {
                "sourcePath":
"$ .Operations",
                "targetPath":
"$ .Operations",

```

```

    nsion:sac:2.0:group-custom-
    parameters']",
      "optional": true
    }
  ]
}

```

```

    "preserveArrayWithSingleElement":
    true,
      "scope": "patchEntity"
    },
    {
      "sourcePath":
      "$.schemas",
      "targetPath":
      "$.schemas",
      "preserveArrayWithSingleElement":
      true,
        "scope": "patchEntity"
      },
      {
        "sourcePath":
        "$.displayName",
        "targetPath": "$.id",
        "scope": "createEntity",
        "functions": [
          {
            "condition":
            "( '%sac.group.prefix%' !=
            'null') && (@ =~ /
            %sac.group.prefix%.*/)",
            "function":
            "replaceFirstString",
            "regex":
            "%sac.group.prefix%",
            "replacement": ""
          }
        ]
      },
      {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.displayName",
        "functions": [
          {
            "condition":
            "( '%sac.group.prefix%' !=
            'null') && (@ =~ /
            %sac.group.prefix%.*/)",
            "function":
            "replaceFirstString",
            "regex":
            "%sac.group.prefix%",
            "replacement": ""
          }
        ]
      },
      {
        "sourcePath":
        "$.externalId",
        "optional": true,
        "targetPath":
        "$.externalId"
      },
      {
        "sourcePath": "$.roles",

```



```

"preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath": "$.roles"
  },
  {
    "sourcePath": "$.members",

"preserveArrayWithSingleElement":
true,
    "targetPath": "$.members",
    "optional": true
  },
  {
    "sourcePath": "$
['urn:sap:params:scim:schemas:exte
nsion:sac:2.0:group-roles']",
    "optional": true,
    "targetPath": "$
['urn:sap:params:scim:schemas:exte
nsion:sac:2.0:group-roles']"
  },
  {
    "sourcePath": "$
['urn:sap:params:scim:schemas:exte
nsion:sac:2.0:group-custom-
parameters']",
    "optional": true,
    "targetPath": "$
['urn:sap:params:scim:schemas:exte
nsion:sac:2.0:group-custom-
parameters']"
  },
  {
    "sourcePath": "$
['urn:sap:cloud:scim:schemas:exten
sion:custom:2.0:Group']
['description']",
    "optional": true,
    "targetPath": "$
['urn:sap:params:scim:schemas:exte
nsion:sac:2.0:group-custom-
parameters']['description']"
  }
]
}
}

```

### ⚠ Caution

When provisioning users and groups between SAP Analytics Cloud exposed as a proxy system and an external system, such as SAP Identity Management, groups are mapped to teams in SAP Analytics Cloud. Those teams can then get role assignments in SAP Analytics Cloud.

If in the external system you run an update job (in the case of SAP Identity Management this is a modify process), role assignments of SAP Analytics Cloud teams will be removed as a result of the PUT operation being executed. This behavior (causing permission issues for users) is expected, as

SAP Analytics Cloud role assignments are not available as group parameters in some external identity management systems. To avoid this, you need to change the [Write Transformation](#) of the [SAP Analytics Cloud](#) proxy system, as described in SAP Note [3027079](#).

### Note

Updating a user in SAP Analytics Cloud using SCIM API version 2 depends on whether user attributes in SAP Analytics Cloud are mapped to SAML attributes in your identity provider. If this is the case, the values of those attributes are populated by the identity provider and cannot be changed by the SAP Analytics SCIM API or the UI. For more information, see [Map SAML Attributes to Users](#)

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"><li>• For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li><li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li></ul>	<ul style="list-style-type: none"><li>• For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li><li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li></ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

8. Run an initial load job.

If you try to provision groups that already exist in SAP Analytics Cloud proxy system, your provisioning job may fail with: *'The group already exists on the target system and cannot be provisioned'* error. This happens when you create a new proxy system and connect it to an existing SAP Analytics Cloud backend.

To avoid this, you have the following options:

- Delete the existing group in SAP Analytics Cloud proxy system.
- Adapt the SAP Analytics Cloud write transformation. Either ignore provisioning of groups or add temporary the [skipOperations](#) expression for creating groups.
- Avoid provisioning of already existing groups.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.6 SAP Application Server ABAP

Follow this procedure to set up SAP Application Server ABAP (AS ABAP) as a proxy system.

## Prerequisites

### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)

- You have credentials of a technical user with read and write permissions for AS ABAP, which plays both the role of a user data source and a target system. Via this user, the Identity Provisioning service will call the ABAP public API in order to execute a number of function modules. These function modules are listed in **step 1** from the procedure below.
- You have the following role, which provides all authorizations with read and write access to user data:  
**SAP\_BC\_JSF\_COMMUNICATION**  
For more information, see: [Configuring the UME to Use an AS ABAP as Data Source](#)


## Context

SAP Application Server ABAP (AS ABAP) offers a user store and user administration capabilities for maintaining users and their authorizations for AS ABAP applications. You can configure AS ABAP as a proxy system so as to provision entities between AS ABAP and another on-premise system.

### i Note

During the [Initial Load](#), only active ABAP users are read. That means, users that have been created before the initial load job is started, and whose expiration date is after the end of the job. To learn about initial load, see **step 8** below – Transformations.

## SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names ([<schema>:<attribute>](#)) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

## ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.ADDRESS.E_MAIL",
  "targetPath": "$.emails[0].value",
  "optional": true,
  "correlationAttribute": true
},
```

You can also set the following filter in the *Properties* tab: `abap.user.membership.filter = ^new.*`

Thus if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=emails[0].value eq "john.smith03@dummymail.com"**

The query request to the SAP AS ABAP BAPI will result into a search in BAPI\_USER\_GETLIST → ADDRESS → *E\_MAIL* field.

This way, the filter will search for a user with an e-mail address '*john.smith03@dummymail.com*' among all users whose ABAP role starts with '*new*'.

## Procedure

1. Open Cloud Connector to add an access control system mapping for **AS ABAP**. This is needed to allow the Identity Provisioning service to access AS ABAP as a back-end system on the intranet. To learn how, see: [Configure Access Control \(RFC\)](#)

Go to ► *Cloud To On-Premise* ► *Access Control* ► tab and select protocol *RFC SNC*. Then, expose the following *exact names* as accessible resources:

- PRGN\_ROLE\_GETLIST
- BAPI\_USER\_GETLIST
- BAPI\_USER\_GET\_DETAIL
- BAPI\_USER\_CREATE1
- BAPI\_USER\_ACTGROUPS\_ASSIGN
- IDENTITY\_MODIFY
- BAPI\_USER\_DELETE
- PRGN\_ACTIVITY\_GROUPS\_LOAD\_RFC

2. Open SAP BTP cockpit, and in your Identity Provisioning subaccount create a destination for the AS ABAP system. To learn how, see: [Create RFC Destinations](#)

The destination configuration is required by the Identity Provisioning service to find the back-end system to be used for reading data. It also provides the credentials of the technical user, needed for the connection to the ABAP public API.

Below are the fields you have to fill in the cockpit destination before using an AS ABAP client as a target system:

Field/Property Name	Value
<i>Name</i>	Enter a destination name.
<i>Type</i>	Select <i>RFC</i> .
<i>User</i>	Enter the user for AS ABAP.  The <i>User</i> field corresponds to property <code>jco.client.user</code> in the exported RFC destination.
<i>Password</i>	(Credential) Enter the password for the AS ABAP user.  The <i>Password</i> field corresponds to property <code>jco.client.passwd</code> in the exported RFC destination.
<code>jco.client.client</code>	Provide the client to be used in the ABAP system. Valid format is a three-digit number.
<code>jco.destination.proxy_type</code>	Defines the proxy type of the connection you need to provide for your ABAP system.  The proxy type <i>OnPremise</i> requires the Cloud Connector to access resources within your on-premise network.  Enter: <i>OnPremise</i>
<b>Direct Connection</b>	
<code>jco.client.ashost</code>	Provide the virtual host entry that you have configured in the Cloud Connector → <i>Access Control</i> configuration.
<code>jco.client.sysnr</code>	Provide the "system number" of the ABAP system.
<b>Load Balancing Connection</b>	
<code>jco.client.mshost</code>	Represents the message server host to be used.
<code>jco.client.r3name</code>	Provide the three-character system ID of the ABAP system to be addressed.
<code>jco.client.msservt</code>	Provide the port on which the message server is listening for incoming requests. You can use this property as an alternative to <code>jco.client.r3name</code> .
<b>Optional Properties</b>	
<code>jco.destination.peak_limit</code>	The value represents the maximum number of active connections that can simultaneously be created for a destination. For example: <i>10</i>
<code>jco.destination.pool_capacity</code>	The value represents the maximum number of idle connections kept open by the destination. For example: <i>5</i>

Field/Property Name	Value
<code>abap.user.name.filter</code>	<p>Filters user names by a regular expression. The regex can define any kind of search pattern.</p> <p>For example, <code>abap.user.name.filter = ^MAR.*</code> reads all user names that start with <i>MAR</i>, such as <b>MARK</b>, <b>MARTINA</b>, and so on.</p> <div> <p><b>i Note</b></p> <p>This property has a higher priority over <code>abap.user.filter</code>. That means, if you set both properties in a system, the value of <b>abap.user.name.filter</b> will be used. However, if the value of <code>abap.user.name.filter</code> is empty, then <b>abap.user.filter</b>'s value will be used instead.</p> </div>
<code>abap.role.name.filter</code>	<p>Filters user roles by a regular expression. The regex can define any kind of search pattern.</p> <p>For example, <code>abap.role.name.filter = ^inter.*</code> reads all users who have roles which start with <i>inter</i>, such as <b>internal</b>, <b>internship</b>, and so on.</p> <div> <p><b>i Note</b></p> <p>This property has a higher priority over <code>abap.role.filter</code>. That means, if you set both properties in a system, the value of <b>abap.role.name.filter</b> will be used. However, if the value of <code>abap.role.name.filter</code> is empty, then <b>abap.role.filter</b>'s value will be used instead.</p> </div>
<code>abap.user.membership.filter</code>	<p>Filters users by a regular expression, based on their <i>Role</i> memberships in AS ABAP. The regex can define any kind of search pattern.</p> <p>For example, <code>abap.user.membership.filter = (?i)^new.*</code> reads all users who have an assigned role which starts with <i>new</i>. This regex is case insensitive, which means the result can be roles starting with <b>new</b>, or <b>New</b>, or <b>NEW</b>, and so on.</p> <div> <p><b>i Note</b></p> <p>If connection properties, like <code>User</code> and <code>Password</code>, are configured both in the destination (SAP BTP cockpit) and on the <i>Properties</i> tab (Identity Provisioning User Interface), the values set in the destination are considered with higher priority.</p> </div>

3. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <i>Users &amp; Authorizations</i> ► <i>Administrators</i> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <i>Access Proxy System API</i> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <i>Security</i> ► <i>OAuth</i> ► <i>Clients</i> ► and choose <i>Register New Client</i>.</li> <li>2. From the <i>Subscription</i> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <i>Authorization Grant</i> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <i>Secret</i> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <i>Client ID</i>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <i>Subscriptions</i> ► <i>Java Applications</i> ► <i>ipsproxy</i> ►.</li> <li>7. From the left-side navigation, choose ► <i>Roles</i> ► <i>IPS_PROXY_USER</i> ►.</li> <li>8. Choose <i>Assign</i> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

4. Access Identity Provisioning. See: [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
5. Add *SAP Application Server ABAP* as a proxy system. To learn how, see: [Add a System \[page 1477\]](#)
6. From the *Destination Name* dropdown, choose the RFC destination you have created in [step 2](#).
7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Application Server ABAP* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your AS ABAP. For more information, see [Manage Transformations \[page 1494\]](#).

Default read and write transformations:

#### → Tip


The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users*



or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

- Use the default transformations if the version of your AS ABAP system is *7.40* or higher. If your system version is *7.31*, you can still use these transformations – just apply SAP Note [1695883](#) .
- NOTE:** If a user is *inactive* in the identity management system, the default Write Transformation creates it as *inactive* (locked) in AS ABAP too.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath":
        "$.USERNAME",
        "targetVariable":
        "entityIdSourceSystem",
        "targetPath": "$.id",
        "functions": [
          {
            "type": "encode",
            "algorithm":
            "base32",
            "skipPadding": true
          }
        ]
      },
      {
        "sourcePath":
        "$.USERNAME",
        "targetPath":
        "$.userName",
        "correlationAttribute":
        true
      },
      {
        "sourcePath":
        "$.ALIAS.USERALIAS",
        "optional": true,
        "targetPath":
        "$.externalId",
        "correlationAttribute":
        true
      },
      {
        "constant":
        "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
        "$.schemas[0]"
      },
      {
        "constant": "User",
        "targetPath":
        "$.meta.resourceType"
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath":
        "$.userName",
        "targetPath":
        "$.USERNAME",
        "targetVariable":
        "entityIdTargetSystem"
      },
      {
        "scope": "deleteEntity",
        "sourceVariable":
        "entityIdTargetSystem",
        "targetVariable":
        "entityIdTargetSystem",
        "functions": [
          {
            "type": "decode",
            "algorithm":
            "base32",
            "skipPadding": true
          },
          {
            "type": "toString"
          }
        ]
      },
      {
        "sourcePath":
        "$.externalId",
        "optional": true,
        "targetPath":
        "$.ALIAS.USERALIAS"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
[ 'userUuid' ]",
        "optional": true,
        "targetPath":
        "$.SAPUSER_UUID.SAP_UID"
      },
      {
        "condition": "$.emails[?
(@.primary == true)].value !=
[]",
        "sourcePath":
        "$.emails[?(@.primary ==
true)].value",
        "preserveArrayWithSingleElement"
        : false,
        "optional": true,
        "targetPath":
        "$.ADDRESS.E_MAIL"

```

```

        "type":
"concatString",
        "suffix": "$
{entityIdSourceSystem}"
    }
    ],
    {
        "sourcePath":
"$$.ADDRESS.E_MAIL",
        "targetPath":
"$$.emails[0].value",
        "optional": true,
        "correlationAttribute":
true
    },
    {
        "condition":
"$$.ADDRESS.E_MAIL EMPTY false",
        "constant": true,
        "targetPath":
"$$.emails[0].primary"
    },
    {
        "condition":
"$$.ADDRESS.E_MAIL EMPTY false",
        "constant": "work",
        "targetPath":
"$$.emails[0].type"
    },
    {
        "sourcePath":
"$$.ADDRESS.FIRSTNAME",
        "optional": true,
        "targetPath":
"$$.name.givenName"
    },
    {
        "sourcePath":
"$$.ADDRESS.LASTNAME",
        "optional": true,
        "targetPath":
"$$.name.familyName"
    },
    {
        "sourcePath":
"$$.ADDRESS.MIDDLENAME",
        "optional": true,
        "targetPath":
"$$.name.middleName"
    },
    {
        "sourcePath":
"$$.ADDRESS.NICKNAME",
        "optional": true,
        "targetPath":
"$$.nickName"
    },
    {
        "sourcePath":
"$$.ADDRESS.TITLE_P",
        "optional": true,

```

```

    },
    {
        "condition": "$.emails[?
(@.type == 'work')].value !=
[]",
        "sourcePath":
"$$.emails[?(@.type ==
'work')].value",
        "preserveArrayWithSingleElement"
: true,
        "optional": true,
        "targetPath":
"$$.ADDRESS.E_MAIL",
        "functions": [
            {
                "function":
"elementAt",
                "index": 0
            }
        ],
        "sourcePath":
"$$.emails[*].value",
        "preserveArrayWithSingleElement"
: true,
        "optional": true,
        "targetPath":
"$$.ADDSMTP[?(@.E_MAIL)]"
    },
    {
        "sourcePath":
"$$.name.givenName",
        "optional": true,
        "targetPath":
"$$.ADDRESS.FIRSTNAME"
    },
    {
        "sourcePath":
"$$.name.familyName",
        "optional": true,
        "targetPath":
"$$.ADDRESS.LASTNAME"
    },
    {
        "scope": "createEntity",
        "sourcePath":
"$$.name.familyName",
        "targetPath":
"$$.ADDRESS.LASTNAME"
    },
    {
        "sourcePath":
"$$.name.middleName",
        "optional": true,
        "targetPath":
"$$.ADDRESS.MIDDLENAME"
    },
    {
        "sourcePath":
"$$.nickName",

```

## Read Transformation

```

      "targetPath":
"$ .name.honorificPrefix"
    },
    {
      "sourcePath":
"$ .ADDRESS.COUNTRY",
      "optional": true,
      "targetPath":
"$ .addresses[0].country"
    },
    {
      "condition":
"$ .ADDRESS.COUNTRY EMPTY false",
      "constant": true,
      "targetPath":
"$ .addresses[0].primary"
    },
    {
      "condition":
"$ .ADDRESS.COUNTRY EMPTY false",
      "constant": "work",
      "targetPath":
"$ .addresses[0].type"
    },
    {
      "sourcePath":
"$ .ADDRESS.TEL1_NUMBR",
      "optional": true,
      "targetPath":
"$ .phoneNumbers[0].value"
    },
    {
      "condition":
"$ .ADDRESS.TEL1_NUMBR EMPTY
false",
      "constant": true,
      "targetPath":
"$ .phoneNumbers[0].primary"
    },
    {
      "condition":
"$ .ADDRESS.TEL1_NUMBR EMPTY
false",
      "constant": "work",
      "targetPath":
"$ .phoneNumbers[0].type"
    },
    {
      "type": "valueMapping",
      "sourcePaths": [
        "$ .DEFAULTS.LANGU"
      ],
      "optional": true,
      "targetPath":
"$ .locale",
      "defaultValue": "en",
      "valueMappings": [
        {
          "key": [
            "W"
          ],
          "mappedValue": "bg"

```

## Write Transformation

```

      "optional": true,
      "targetPath":
"$ .ADDRESS.NICKNAME"
    },
    {
      "sourcePath":
"$ .name.honorificPrefix",
      "optional": true,
      "targetPath":
"$ .ADDRESS.TITLE_P"
    },
    {
      "condition":
"$ .phoneNumbers[?(@.primary ==
true)].value != []",
      "sourcePath":
"$ .phoneNumbers[?(@.primary ==
true)].value",
      "preserveArrayWithSingleElement"
: false,
      "optional": true,
      "targetPath":
"$ .ADDRESS.TEL1_NUMBR"
    },
    {
      "condition":
"$ .phoneNumbers[?(@.type ==
'work')].value != []",
      "sourcePath":
"$ .phoneNumbers[?(@.type ==
'work')].value",
      "preserveArrayWithSingleElement"
: true,
      "optional": true,
      "targetPath":
"$ .ADDRESS.TEL1_NUMBR",
      "functions": [
        {
          "function":
"elementAt",
          "index": 0
        }
      ],
      "sourcePath":
"$ .phoneNumbers[*].value",
      "preserveArrayWithSingleElement"
: true,
      "optional": true,
      "targetPath":
"$ .ADDTTEL[?(@.TELEPHONE)]"
    },
    {
      "type": "valueMapping",
      "sourcePaths": [
        "$ .locale"
      ],
      "optional": true,

```

```

    }
  ],
  {
    "type": "valueMapping",
    "sourcePaths": [
      "$.ADDRESS.LANGUP_ISO"
    ],
    "optional": true,
    "targetPath":
    "$.preferredLanguage",
    "functions": [
      {
        "function":
        "toLowerCaseString"
      }
    ],
  },
  {
    "type": "valueMapping",
    "sourcePaths": [
      "$.LOGONDATA.TZONE"
    ],
    "optional": true,
    "targetPath":
    "$.timezone",
    "defaultValue": "Europe/
    Berlin",
    "valueMappings": [
      {
        "key": [
          "EET"
        ],
        "mappedValue":
        "Europe/Sofia"
      }
    ],
  },
  {
    "constant": false,
    "targetPath": "$.active"
  },
  {
    "condition":
    "($.ISLOCKED.LOCAL_LOCK != 'L')
    && ($.ISLOCKED.NO_USER_PW !=
    'L') && ($.ISLOCKED.GLOB_LOCK !
    = 'L') &&
    ($.ISLOCKED.WRNG_LOGON != 'L')
    && ($.LOCK != 'L') &&
    ($.LOCK_LOCALLY != 'X')",
    "constant": true,
    "targetPath": "$.active"
  },
  {
    "sourcePath":
    "$.ACTIVITYGROUPS[*].AGR_NAME",
    "preserveArrayWithSingleElement"
    : true,
    "optional": true,

```

```

    "targetPath":
    "$.DEFAULTS.LANGU",
    "defaultValue": "E",
    "valueMappings": [
      {
        "key": [
          "bg"
        ],
        "mappedValue": "W"
      }
    ],
  },
  {
    "sourcePath":
    "$.preferredLanguage",
    "optional": true,
    "targetPath":
    "$.ADDRESS.LANGUP_ISO",
    "functions": [
      {
        "function":
        "toUpperCaseString"
      }
    ],
  },
  {
    "type": "valueMapping",
    "sourcePaths": [
      "$.timezone"
    ],
    "optional": true,
    "targetPath":
    "$.LOGONDATA.TZONE",
    "defaultValue": "CET",
    "valueMappings": [
      {
        "key": [
          "Europe/Sofia"
        ],
        "mappedValue": "EET"
      }
    ],
  },
  {
    "scope": "createEntity",
    "targetPath":
    "$.PASSWORD.BAPIPWD",
    "functions": [
      {
        "type":
        "randomPassword",
        "passwordLength":
        24,
        "minimumNumberOfLowercaseLetters
        ": 1,
        "minimumNumberOfUppercaseLetters
        ": 1,
        "minimumNumberOfDigits": 1,

```

```

        "targetPath":
"$ .groups[?(@.value)]",
        "functions": [
            {
                "type": "encode",
                "algorithm":
"base32",
                "skipPadding": true
            }
        ],
        "constant": "direct",

"preserveArrayWithSingleElement"
: true,
        "targetPath":
"$ .groups[*].type",
        "optional": true
    }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "sourcePath":
"$ .ROLE_NAME",
                "targetVariable":
"entityIdSourceSystem",
                "targetPath": "$ .id",
                "functions": [
                    {
                        "type": "encode",
                        "algorithm":
"base32",
                        "skipPadding": true
                    }
                ],
                "constant": "Group",
                "targetPath":
"$ .meta.resourceType"
            },
            {
                "sourceVariable":
"entityBaseLocation",
                "targetVariable":
"entityLocationSourceSystem",
                "targetPath":
"$ .meta.location",
                "functions": [
                    {
                        "type":
"concatString",
                        "suffix": "$
{entityIdSourceSystem}"
                    }
                ]
            }
        ]
    },

```

```

"minimumNumberOfSpecialSymbols":
0
    }
    ],
    {
        "scope": "createEntity",
        "optional": true,
        "ignore": true,
        "sourcePath":
"$ .password",
        "targetPath":
"$ .PASSWORD.BAPIPWD"
    },
    {
        "constant":
"updateEntity",
        "targetVariable":
"operationTypeVariable"
    },
    {
        "constant":
"createEntity",
        "targetVariable":
"operationTypeVariable",
        "scope": "createEntity"
    },
    {
        "condition":
"$ .active == false && '$
{operationTypeVariable}' ==
'createEntity'",
        "constant": "X",
        "targetPath":
"$ .LOCK_LOCALLY"
    },
    {
        "condition": "'$
{operationTypeVariable}' ==
'updateEntity'",
        "constant": "U",
        "targetPath": "$ .LOCK"
    },
    {
        "condition":
"$ .active == false && '$
{operationTypeVariable}' ==
'updateEntity'",
        "constant": "L",
        "targetPath": "$ .LOCK"
    }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "sourcePath":
"$ .displayName",
                "targetPath":
"$ .ROLE_NAME"
            }
        ]
    }

```

## Read Transformation

```

    {
      "sourcePath":
        "$.ROLE_NAME",
      "targetPath":
        "$.displayName"
    },
    {
      "constant":
        "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath":
        "$.schemas[0]"
    },
    {
      "sourcePath":
        "$.USERLIST[*].USERNAME",
      "preserveArrayWithSingleElement":
        true,
      "targetPath":
        "$.members[?(@.value)]",
      "optional": true,
      "functions": [
        {
          "type": "encode",
          "algorithm":
            "base32",
          "skipPadding": true
        }
      ]
    },
    {
      "constant": "User",
      "preserveArrayWithSingleElement":
        true,
      "targetPath":
        "$.members[*].type",
      "optional": true
    }
  ]
}

```

## Write Transformation

```

    },
    {
      "sourceVariable":
        "entityIdTargetSystem",
      "targetVariable":
        "entityIdTargetSystem",
      "functions": [
        {
          "type": "decode",
          "algorithm":
            "base32",
          "skipPadding": true
        },
        {
          "type": "toString"
        }
      ]
    },
    {
      "sourcePath":
        "$.members[*].value",
      "preserveArrayWithSingleElement":
        true,
      "targetPath":
        "$.USERLIST[?(@.USERNAME)]",
      "optional": true,
      "functions": [
        {
          "type": "decode",
          "algorithm":
            "base32",
          "skipPadding": true
        },
        {
          "type": "toString",
          "applyOnElements":
            true
        }
      ]
    }
  ]
}

```

- If your AS ABAP version is [7.30](#) or lower, use the transformations below.

**NOTE:** The Write Transformation creates all users as *active* (unlocked) in AS ABAP, regardless if they are *active* or *inactive* in the identity management system.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath":
        "$.USERNAME",
        "targetVariable":
        "entityIdSourceSystem",
        "targetPath": "$.id",
        "functions": [
          {
            "type": "encode",
            "algorithm":
            "base32",
            "skipPadding": true
          }
        ],
        "sourcePath":
        "$.USERNAME",
        "targetPath":
        "$.userName",
        "correlationAttribute":
        true
      },
      {
        "sourcePath":
        "$.ALIAS.USERALIAS",
        "optional": true,
        "targetPath":
        "$.externalId",
        "correlationAttribute":
        true
      },
      {
        "constant":
        "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
        "$.schemas[0]"
      },
      {
        "constant": "User",
        "targetPath":
        "$.meta.resourceType"
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath":
        "$.userName",
        "targetPath":
        "$.USERNAME",
        "targetVariable":
        "entityIdTargetSystem"
      },
      {
        "scope": "deleteEntity",
        "sourceVariable":
        "entityIdTargetSystem",
        "targetVariable":
        "entityIdTargetSystem",
        "functions": [
          {
            "type": "decode",
            "algorithm":
            "base32",
            "skipPadding": true
          },
          {
            "type": "toString"
          }
        ],
        "sourcePath":
        "$.externalId",
        "optional": true,
        "targetPath":
        "$.ALIAS.USERALIAS"
      },
      {
        "condition": "$.emails[?
(@.primary == true)].value !=
[]",
        "sourcePath":
        "$.emails[?(@.primary ==
true)].value",
        "preserveArrayWithSingleElement"
        : false,
        "optional": true,
        "targetPath":
        "$.ADDRESS.E_MAIL"
      },
      {
        "condition": "$.emails[?
(@.type == 'work')].value !=
[]",
        "sourcePath":
        "$.emails[?(@.type ==
'work')].value",

```



```

      "type":
      "concatString",
      "suffix": "$
    {entityIdSourceSystem}"
    }
  ],
  {
    "sourcePath":
    "$.ADDRESS.E_MAIL",
    "targetPath":
    "$.emails[0].value",
    "optional": true,
    "correlationAttribute":
    true
  },
  {
    "condition":
    "$.ADDRESS.E_MAIL EMPTY false",
    "constant": true,
    "targetPath":
    "$.emails[0].primary"
  },
  {
    "condition":
    "$.ADDRESS.E_MAIL EMPTY false",
    "constant": "work",
    "targetPath":
    "$.emails[0].type"
  },
  {
    "sourcePath":
    "$.ADDRESS.FIRSTNAME",
    "optional": true,
    "targetPath":
    "$.name.givenName"
  },
  {
    "sourcePath":
    "$.ADDRESS.LASTNAME",
    "optional": true,
    "targetPath":
    "$.name.familyName"
  },
  {
    "sourcePath":
    "$.ADDRESS.MIDDLENAME",
    "optional": true,
    "targetPath":
    "$.name.middleName"
  },
  {
    "sourcePath":
    "$.ADDRESS.NICKNAME",
    "optional": true,
    "targetPath":
    "$.nickName"
  },
  {
    "sourcePath":
    "$.ADDRESS.TITLE_P",
    "optional": true,

```

```

"preserveArrayWithSingleElement"
: true,
  "optional": true,
  "targetPath":
  "$.ADDRESS.E_MAIL",
  "functions": [
    {
      "function":
      "elementAt",
      "index": 0
    }
  ],
  {
    "sourcePath":
    "$.emails[*].value",
    "preserveArrayWithSingleElement"
    : true,
    "optional": true,
    "targetPath":
    "$.ADDSMTP[?(@.E_MAIL)]"
  },
  {
    "sourcePath":
    "$.name.givenName",
    "optional": true,
    "targetPath":
    "$.ADDRESS.FIRSTNAME"
  },
  {
    "sourcePath":
    "$.name.familyName",
    "optional": true,
    "targetPath":
    "$.ADDRESS.LASTNAME"
  },
  {
    "scope": "createEntity",
    "sourcePath":
    "$.name.familyName",
    "targetPath":
    "$.ADDRESS.LASTNAME"
  },
  {
    "sourcePath":
    "$.name.middleName",
    "optional": true,
    "targetPath":
    "$.ADDRESS.MIDDLENAME"
  },
  {
    "sourcePath":
    "$.nickName",
    "optional": true,
    "targetPath":
    "$.ADDRESS.NICKNAME"
  },
  {
    "sourcePath":
    "$.name.honorificPrefix",
    "optional": true,

```

```

        "targetPath":
        "$.name.honorificPrefix"
        },
        {
            "sourcePath":
            "$.ADDRESS.COUNTRY",
            "optional": true,
            "targetPath":
            "$.addresses[0].country"
        },
        {
            "condition":
            "$.ADDRESS.COUNTRY EMPTY false",
            "constant": true,
            "targetPath":
            "$.addresses[0].primary"
        },
        {
            "condition":
            "$.ADDRESS.COUNTRY EMPTY false",
            "constant": "work",
            "targetPath":
            "$.addresses[0].type"
        },
        {
            "sourcePath":
            "$.ADDRESS.TEL1_NUMBR",
            "optional": true,
            "targetPath":
            "$.phoneNumbers[0].value"
        },
        {
            "condition":
            "$.ADDRESS.TEL1_NUMBR EMPTY
            false",
            "constant": true,
            "targetPath":
            "$.phoneNumbers[0].primary"
        },
        {
            "condition":
            "$.ADDRESS.TEL1_NUMBR EMPTY
            false",
            "constant": "work",
            "targetPath":
            "$.phoneNumbers[0].type"
        },
        {
            "type": "valueMapping",
            "sourcePaths": [
                "$.DEFAULTS.LANGU"
            ],
            "optional": true,
            "targetPath":
            "$.locale",
            "defaultValue": "en",
            "valueMappings": [
                {
                    "key": [
                        "W"
                    ],
                    "mappedValue": "bg"
                }
            ]
        }
    ]
}

```

```

        "targetPath":
        "$.ADDRESS.TITLE_P"
        },
        {
            "condition":
            "$.phoneNumbers[?(@.primary ==
            true)].value != []",
            "sourcePath":
            "$.phoneNumbers[?(@.primary ==
            true)].value",
            "preserveArrayWithSingleElement"
            : false,
            "optional": true,
            "targetPath":
            "$.ADDRESS.TEL1_NUMBR"
        },
        {
            "condition":
            "$.phoneNumbers[?(@.type ==
            'work')].value != []",
            "sourcePath":
            "$.phoneNumbers[?(@.type ==
            'work')].value",
            "preserveArrayWithSingleElement"
            : true,
            "optional": true,
            "targetPath":
            "$.ADDRESS.TEL1_NUMBR",
            "functions": [
                {
                    "function":
                    "elementAt",
                    "index": 0
                }
            ],
            "sourcePath":
            "$.phoneNumbers[*].value",
            "preserveArrayWithSingleElement"
            : true,
            "optional": true,
            "targetPath":
            "$.ADDTTEL[?(@.TELEPHONE)]"
        },
        {
            "type": "valueMapping",
            "sourcePaths": [
                "$.locale"
            ],
            "optional": true,
            "targetPath":
            "$.DEFAULTS.LANGU",
            "defaultValue": "E",
            "valueMappings": [
                {
                    "key": [
                        "bg"
                    ],
                    "mappedValue": "W"
                }
            ]
        }
    ]
}

```

```

    }
  ],
  {
    "type": "valueMapping",
    "sourcePaths": [
      "$.ADDRESS.LANGUP_ISO"
    ],
    "optional": true,
    "targetPath":
    "$.preferredLanguage",
    "functions": [
      {
        "function":
        "toLowerCaseString"
      }
    ]
  },
  {
    "type": "valueMapping",
    "sourcePaths": [
      "$.LOGONDATA.TZONE"
    ],
    "optional": true,
    "targetPath":
    "$.timezone",
    "defaultValue": "Europe/
    Berlin",
    "valueMappings": [
      {
        "key": [
          "EET"
        ],
        "mappedValue":
        "Europe/Sofia"
      }
    ]
  },
  {
    "constant": false,
    "targetPath": "$.active"
  },
  {
    "condition":
    "($.ISLOCKED.LOCAL_LOCK != 'L')
    && ($.ISLOCKED.NO_USER_PW !=
    'L') && ($.ISLOCKED.GLOB_LOCK !
    = 'L') &&
    ($.ISLOCKED.WRNG_LOGON != 'L')
    && ($.LOCK != 'L') &&
    ($.LOCK_LOCALLY != 'X')",
    "constant": true,
    "targetPath": "$.active"
  },
  {
    "sourcePath":
    "$.ACTIVITYGROUPS[*].AGR_NAME",
    "preserveArrayWithSingleElement"
    : true,
    "optional": true,

```

```

    }
  ],
  {
    "sourcePath":
    "$.preferredLanguage",
    "optional": true,
    "targetPath":
    "$.ADDRESS.LANGUP_ISO",
    "functions": [
      {
        "function":
        "toUpperCaseString"
      }
    ]
  },
  {
    "type": "valueMapping",
    "sourcePaths": [
      "$.timezone"
    ],
    "optional": true,
    "targetPath":
    "$.LOGONDATA.TZONE",
    "defaultValue": "CET",
    "valueMappings": [
      {
        "key": [
          "Europe/Sofia"
        ],
        "mappedValue": "EET"
      }
    ]
  },
  {
    "scope": "createEntity",
    "targetPath":
    "$.PASSWORD.BAPIPWD",
    "functions": [
      {
        "type":
        "randomPassword",
        "passwordLength":
        24,
        "minimumNumberOfLowercaseLetters
        ": 1,
        "minimumNumberOfUppercaseLetters
        ": 1,
        "minimumNumberOfDigits": 1,
        "minimumNumberOfSpecialSymbols":
        0
      }
    ]
  },
  {
    "scope": "createEntity",
    "optional": true,
    "ignore": true,

```

```

        "targetPath":
"$ .groups[?(@.value)]",
        "functions": [
            {
                "type": "encode",
                "algorithm":
"base32",
                "skipPadding": true
            }
        ],
        {
            "constant": "direct",

"preserveArrayWithSingleElement"
: true,
            "targetPath":
"$ .groups[*].type",
            "optional": true
        }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "sourcePath":
"$ .ROLE_NAME",
                "targetVariable":
"entityIdSourceSystem",
                "targetPath": "$ .id",
                "functions": [
                    {
                        "type": "encode",
                        "algorithm":
"base32",
                        "skipPadding": true
                    }
                ],
                {
                    "constant": "Group",
                    "targetPath":
"$ .meta.resourceType"
                },
                {
                    "sourceVariable":
"entityBaseLocation",
                    "targetVariable":
"entityLocationSourceSystem",
                    "targetPath":
"$ .meta.location",
                    "functions": [
                        {
                            "type":
"concatString",
                            "suffix": "$
{entityIdSourceSystem}"
                        }
                    ]
                }
            ],
        },
    },

```

```

        "sourcePath":
"$ .password",
        "targetPath":
"$ .PASSWORD.BAPIPWD"
    },
    {
        "scope": "createEntity",
        "sourcePath":
"$ .groups[*].value",

"preserveArrayWithSingleElement"
: true,
        "optional": true,
        "targetPath":
"$ .ACTIVITYGROUPS[?
(@.AGR_NAME)]",
        "functions": [
            {
                "type": "decode",
                "algorithm":
"base32",
                "skipPadding": true
            },
            {
                "type": "toString",
                "applyOnElements":
true
            }
        ],
        {
            "constant":
"updateEntity",
            "targetVariable":
"operationTypeVariable"
        },
        {
            "constant":
"createEntity",
            "targetVariable":
"operationTypeVariable",
            "scope": "createEntity"
        },
        {
            "condition": "'$
{operationTypeVariable}' ==
'updateEntity'",
            "constant": "U",
            "targetPath": "$ .LOCK"
        },
        {
            "condition":
"$ .active == false && '$
{operationTypeVariable}' ==
'updateEntity'",
            "constant": "L",
            "targetPath": "$ .LOCK"
        }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
    },

```

```

    {
      "sourcePath":
        "$.ROLE_NAME",
      "targetPath":
        "$.displayName"
    },
    {
      "constant":
        "urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath":
        "$.schemas[0]"
    },
    {
      "sourcePath":
        "$.USERLIST[*].USERNAME",
      "preserveArrayWithSingleElement": true,
      "targetPath":
        "$.members[?(@.value)]",
      "optional": true,
      "functions": [
        {
          "type": "encode",
          "algorithm":
            "base32",
          "skipPadding": true
        }
      ],
      "constant": "User",
      "preserveArrayWithSingleElement": true,
      "targetPath":
        "$.members[*].type",
      "optional": true
    }
  ]
}

```

```

    "mappings": [
      {
        "sourcePath":
          "$.displayName",
        "targetPath":
          "$.ROLE_NAME"
      },
      {
        "sourceVariable":
          "entityIdTargetSystem",
        "targetVariable":
          "entityIdTargetSystem",
        "functions": [
          {
            "type": "decode",
            "algorithm":
              "base32",
            "skipPadding": true
          },
          {
            "type": "toString"
          }
        ]
      },
      {
        "sourcePath":
          "$.members[*].value",
        "preserveArrayWithSingleElement": true,
        "targetPath":
          "$.USERLIST[?(@.USERNAME)]",
        "optional": true,
        "functions": [
          {
            "type": "decode",
            "algorithm":
              "base32",
            "skipPadding": true
          },
          {
            "type": "toString",
            "applyOnElements": true
          }
        ]
      }
    ]
  }
}

```

- Default transformations supporting User UUID attribute:

#### Read Transformation

##### ≡ Code Syntax

```
{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath":
        "$.USERNAME",
        "targetVariable":
        "entityIdSourceSystem",
        "targetPath": "$.id",
        "functions": [
          {
            "type": "encode",
            "algorithm":
            "base32",
            "skipPadding": true
          }
        ],
        {
          "sourcePath":
          "$.USERNAME",
          "targetPath":
          "$.userName",
          "correlationAttribute":
          true
        },
        {
          "sourcePath":
          "$.ALIAS.USERALIAS",
          "optional": true,
          "targetPath":
          "$.externalId",
          "correlationAttribute":
          true
        },
        {
          "sourcePath":
          "$.SAPUSER_UUID.SAP_UID",
          "optional": true,
          "targetPath": "$
          ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
          ['userUuid']"
        },
        {
          "constant":
          "urn:ietf:params:scim:schemas:co
          re:2.0:User",
          "targetPath":
          "$.schemas[0]"
        },
        {
          "constant": "User",
```

#### Write Transformation

##### ≡ Code Syntax

```
{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath":
        "$.userName",
        "targetPath":
        "$.USERNAME",
        "targetVariable":
        "entityIdTargetSystem"
      },
      {
        "scope": "deleteEntity",
        "sourceVariable":
        "entityIdTargetSystem",
        "targetVariable":
        "entityIdTargetSystem",
        "functions": [
          {
            "type": "decode",
            "algorithm":
            "base32",
            "skipPadding": true
          },
          {
            "type": "toString"
          }
        ],
        {
          "sourcePath":
          "$.externalId",
          "optional": true,
          "targetPath":
          "$.ALIAS.USERALIAS"
        },
        {
          "sourcePath": "$
          ['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
          ['userUuid']",
          "optional": true,
          "targetPath":
          "$.SAPUSER_UUID.SAP_UID"
        },
        {
          "condition": "$.emails[?
          (@.primary == true)].value !=
          []",
          "sourcePath":
          "$.emails[?(@.primary ==
          true)].value",
          "preserveArrayWithSingleElement"
          : false,
```

```

        "targetPath":
        "$.meta.resourceType"
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [
          {
            "type":
            "concatString",
            "suffix": "$"
            {entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
        "$.ADDRESS.E_MAIL",
        "targetPath":
        "$.emails[0].value",
        "optional": true,
        "correlationAttribute":
        true
      },
      {
        "condition":
        "$.ADDRESS.E_MAIL EMPTY false",
        "constant": true,
        "targetPath":
        "$.emails[0].primary"
      },
      {
        "condition":
        "$.ADDRESS.E_MAIL EMPTY false",
        "constant": "work",
        "targetPath":
        "$.emails[0].type"
      },
      {
        "sourcePath":
        "$.ADDRESS.FIRSTNAME",
        "optional": true,
        "targetPath":
        "$.name.givenName"
      },
      {
        "sourcePath":
        "$.ADDRESS.LASTNAME",
        "optional": true,
        "targetPath":
        "$.name.familyName"
      },
      {
        "sourcePath":
        "$.ADDRESS.MIDDLENAME",
        "optional": true,
        "targetPath":
        "$.name.middleName"

```

```

        "optional": true,
        "targetPath":
        "$.ADDRESS.E_MAIL"
      },
      {
        "condition": "$.emails[?
        (@.type == 'work')].value !=
        []",
        "sourcePath":
        "$.emails[?(@.type ==
        'work')].value",
        "preserveArrayWithSingleElement"
        : true,
        "optional": true,
        "targetPath":
        "$.ADDRESS.E_MAIL",
        "functions": [
          {
            "function":
            "elementAt",
            "index": 0
          }
        ],
        "sourcePath":
        "$.emails[*].value",
        "preserveArrayWithSingleElement"
        : true,
        "optional": true,
        "targetPath":
        "$.ADDSMTP[?(@.E_MAIL)]"
      },
      {
        "sourcePath":
        "$.name.givenName",
        "optional": true,
        "targetPath":
        "$.ADDRESS.FIRSTNAME"
      },
      {
        "sourcePath":
        "$.name.familyName",
        "optional": true,
        "targetPath":
        "$.ADDRESS.LASTNAME"
      },
      {
        "scope": "createEntity",
        "sourcePath":
        "$.name.familyName",
        "targetPath":
        "$.ADDRESS.LASTNAME"
      },
      {
        "sourcePath":
        "$.name.middleName",
        "optional": true,
        "targetPath":
        "$.ADDRESS.MIDDLENAME"
      },

```

```

    },
    {
      "sourcePath":
        "$.ADDRESS.NICKNAME",
      "optional": true,
      "targetPath":
        "$.nickName"
    },
    {
      "sourcePath":
        "$.ADDRESS.TITLE_P",
      "optional": true,
      "targetPath":
        "$.name.honorificPrefix"
    },
    {
      "sourcePath":
        "$.ADDRESS.COUNTRY",
      "optional": true,
      "targetPath":
        "$.addresses[0].country"
    },
    {
      "condition":
        "$.ADDRESS.COUNTRY EMPTY false",
      "constant": true,
      "targetPath":
        "$.addresses[0].primary"
    },
    {
      "condition":
        "$.ADDRESS.COUNTRY EMPTY false",
      "constant": "work",
      "targetPath":
        "$.addresses[0].type"
    },
    {
      "sourcePath":
        "$.ADDRESS.TEL1_NUMBR",
      "optional": true,
      "targetPath":
        "$.phoneNumbers[0].value"
    },
    {
      "condition":
        "$.ADDRESS.TEL1_NUMBR EMPTY false",
      "constant": true,
      "targetPath":
        "$.phoneNumbers[0].primary"
    },
    {
      "condition":
        "$.ADDRESS.TEL1_NUMBR EMPTY false",
      "constant": "work",
      "targetPath":
        "$.phoneNumbers[0].type"
    },
    {
      "type": "valueMapping",
      "sourcePaths": [

```

```

    {
      "sourcePath":
        "$.nickName",
      "optional": true,
      "targetPath":
        "$.ADDRESS.NICKNAME"
    },
    {
      "sourcePath":
        "$.name.honorificPrefix",
      "optional": true,
      "targetPath":
        "$.ADDRESS.TITLE_P"
    },
    {
      "condition":
        "$.phoneNumbers[?(@.primary == true)].value != []",
      "sourcePath":
        "$.phoneNumbers[?(@.primary == true)].value",
      "preserveArrayWithSingleElement": false,
      "optional": true,
      "targetPath":
        "$.ADDRESS.TEL1_NUMBR"
    },
    {
      "condition":
        "$.phoneNumbers[?(@.type == 'work')].value != []",
      "sourcePath":
        "$.phoneNumbers[?(@.type == 'work')].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath":
        "$.ADDRESS.TEL1_NUMBR",
      "functions": [
        {
          "function":
            "elementAt",
            "index": 0
        }
      ],
      "sourcePath":
        "$.phoneNumbers[*].value",
      "preserveArrayWithSingleElement": true,
      "optional": true,
      "targetPath":
        "$.ADDTel[?(@.TELEPHONE)]"
    },
    {
      "type": "valueMapping",
      "sourcePaths": [
        "$.locale"

```



```

        "$.DEFAULTS.LANGU"
    ],
    "optional": true,
    "targetPath":
    "$.locale",
    "defaultValue": "en",
    "valueMappings": [
        {
            "key": [
                "W"
            ],
            "mappedValue": "bg"
        }
    ]
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.ADDRESS.LANGUP_ISO"
    ],
    "optional": true,
    "targetPath":
    "$.preferredLanguage",
    "functions": [
        {
            "function":
            "toLowerCaseString"
        }
    ]
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.LOGONDATA.TZONE"
    ],
    "optional": true,
    "targetPath":
    "$.timezone",
    "defaultValue": "Europe/
Berlin",
    "valueMappings": [
        {
            "key": [
                "EET"
            ],
            "mappedValue":
            "Europe/Sofia"
        }
    ]
},
{
    "constant": false,
    "targetPath": "$.active"
},
{
    "condition":
    "($.ISLOCKED.LOCAL_LOCK != 'L')
    && ($.ISLOCKED.GLOB_LOCK !=
    'L') && ($.ISLOCKED.WRNG_LOGON !
    = 'L') && ($.LOCK != 'L') &&
    ($.LOCK_LOCALLY != 'X')",
    "constant": true,

```

```

    ],
    "optional": true,
    "targetPath":
    "$.DEFAULTS.LANGU",
    "defaultValue": "E",
    "valueMappings": [
        {
            "key": [
                "bg"
            ],
            "mappedValue": "W"
        }
    ]
},
{
    "sourcePath":
    "$.preferredLanguage",
    "optional": true,
    "targetPath":
    "$.ADDRESS.LANGUP_ISO",
    "functions": [
        {
            "function":
            "toUpperCaseString"
        }
    ]
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.timezone"
    ],
    "optional": true,
    "targetPath":
    "$.LOGONDATA.TZONE",
    "defaultValue": "CET",
    "valueMappings": [
        {
            "key": [
                "Europe/Sofia"
            ],
            "mappedValue": "EET"
        }
    ]
},
{
    "scope": "createEntity",
    "targetPath":
    "$.PASSWORD.BAPIPWD",
    "functions": [
        {
            "type":
            "randomPassword",
            "passwordLength":
            24,
            "minimumNumberOfLowercaseLetters
            ": 1,
            "minimumNumberOfUppercaseLetters
            ": 1,
            "minimumNumberOfDigits": 1,

```

```

        "targetPath": "$.active"
      },
      {
        "sourcePath":
        "$.ACTIVITYGROUPS[*].AGR_NAME",
        "preserveArrayWithSingleElement"
        : true,
        "optional": true,
        "targetPath":
        "$.groups[?(@.value)]",
        "functions": [
          {
            "type": "encode",
            "algorithm":
            "base32",
            "skipPadding": true
          }
        ]
      },
      {
        "constant": "direct",
        "preserveArrayWithSingleElement"
        : true,
        "targetPath":
        "$.groups[*].type",
        "optional": true
      }
    ],
    "group": {
      "scimEntityEndpoint":
      "Groups",
      "mappings": [
        {
          "sourcePath":
          "$.ROLE_NAME",
          "targetVariable":
          "entityIdSourceSystem",
          "targetPath": "$.id",
          "functions": [
            {
              "type": "encode",
              "algorithm":
              "base32",
              "skipPadding": true
            }
          ]
        },
        {
          "constant": "Group",
          "targetPath":
          "$.meta.resourceType"
        },
        {
          "sourceVariable":
          "entityBaseLocation",
          "targetVariable":
          "entityLocationSourceSystem",
          "targetPath":
          "$.meta.location",

```

```

"minimumNumberOfSpecialSymbols":
0
      }
    ]
  },
  {
    "scope": "createEntity",
    "optional": true,
    "ignore": true,
    "sourcePath":
    "$.password",
    "targetPath":
    "$.PASSWORD.BAPIPWD"
  },
  {
    "scope": "createEntity",
    "sourcePath":
    "$.groups[*].value",
    "preserveArrayWithSingleElement"
    : true,
    "optional": true,
    "targetPath":
    "$.ACTIVITYGROUPS[?
    (@.AGR_NAME)]",
    "functions": [
      {
        "type": "decode",
        "algorithm":
        "base32",
        "skipPadding": true
      },
      {
        "type": "toString",
        "applyOnElements":
        true
      }
    ]
  },
  {
    "constant":
    "updateEntity",
    "targetVariable":
    "operationTypeVariable"
  },
  {
    "constant":
    "createEntity",
    "targetVariable":
    "operationTypeVariable",
    "scope": "createEntity"
  },
  {
    "condition":
    "$.active == false && '$
    {operationTypeVariable}' ==
    'createEntity'",
    "constant": "X",
    "targetPath":
    "$.LOCK_LOCALLY"
  },
  {

```

```

      "functions": [
        {
          "type":
"concatString",
          "suffix": "$
{entityIdSourceSystem}"
        }
      ],
      {
        "sourcePath":
"$$.ROLE_NAME",
        "targetPath":
"$$.displayName"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:co
re:2.0:Group",
        "targetPath":
"$$.schemas[0]"
      },
      {
        "sourcePath":
"$$.USERLIST[*].USERNAME",
        "preserveArrayWithSingleElement"
: true,
        "targetPath":
"$$.members[?(@.value)]",
        "optional": true,
        "functions": [
          {
            "type": "encode",
            "algorithm":
"base32",
            "skipPadding": true
          }
        ],
        {
          "constant": "User",
          "preserveArrayWithSingleElement"
: true,
          "targetPath":
"$$.members[*].type",
          "optional": true
        }
      ]
    }
  }
}

```

```

      "condition": "'$
{operationTypeVariable}' ==
'updateEntity'",
      "constant": "U",
      "targetPath": "$$.LOCK"
    },
    {
      "condition":
"$$.active == false && '$
{operationTypeVariable}' ==
'updateEntity'",
      "constant": "L",
      "targetPath": "$$.LOCK"
    }
  ],
  "group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.ROLE_NAME"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetVariable":
"entityIdTargetSystem",
        "functions": [
          {
            "type": "decode",
            "algorithm":
"base32",
            "skipPadding": true
          },
          {
            "type": "toString"
          }
        ]
      },
      {
        "sourcePath":
"$$.members[*].value",
        "preserveArrayWithSingleElement"
: true,
        "targetPath":
"$$.USERLIST[?(@.USERNAME)]",
        "optional": true,
        "functions": [
          {
            "type": "decode",
            "algorithm":
"base32",
            "skipPadding": true
          },
          {
            "type": "toString",

```

Read Transformation	Write Transformation
	<pre> true "applyOnElements":   [     {       [         {           }         ]       }     ]   ] </pre>

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.7 SAP Ariba Applications

Follow this procedure to set up SAP Ariba Applications as a proxy system.


## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

1. You have created a client application on [SAP Ariba APIs Portal](#)  that needs to be enabled for Identity Provisioning.

### i Note

If you don't have an account on SAP Ariba Developer Portal, then ask your **Designated Support Contact** (DSC) to submit a [request for an account](#). To find your DSC person, see: [How can I see my company's Basic users and Designated Support Contacts \(DSC\)](#) 

2. Provide your DSC person with your SAP Ariba **realm name**, **application name**, and **application key**. You have already created the application name along with the application key on [step 2](#). To find your realm name, login to your SAP Ariba system – it's part of your login URL, as shown in the following examples.
  - [SAP Ariba Buyer](#) example: `https://s1.ariba.com/Buyer/Main/ad/loginPage/...&realm=mycompany-t`

- *SAP Ariba Sourcing* example: <http://mycompany.sourcing.ariba.com/>
3. Ask your DSC person to submit a service request for you to *SAP Ariba Support* for component **BNS-ARI-SS-API**, requesting the client application to be enabled for Identity Provisioning. Request your DSC person to mention the following details in the service request:
    - Application name
    - Application key
    - Realm name
  4. When your application is enabled, you can login to *SAP Ariba APIs Portal*, find your application, and generate a new OAuth secret for it. To learn how, see: [How to generate the OAuth Secret and Base64 Encoded Client and secret](#)
  5. To configure your *SAP Ariba Applications* provisioning system (see the procedure below), you will need to map your SAP Ariba application parameters to the relevant Identity Provisioning properties. The property mapping between the two systems is as follows:

SAP Ariba	Identity Provisioning	Values
SCIM API URL	URL	Examples: <ul style="list-style-type: none"> <li>• US: <a href="https://openapi.ariba.com">https://openapi.ariba.com</a></li> <li>• Europe: <a href="https://eu.openapi.ariba.com">https://eu.openapi.ariba.com</a></li> <li>• UAE: <a href="https://mn1.openapi.ariba.com">https://mn1.openapi.ariba.com</a></li> </ul>
SAP Ariba OAuth 2.0 Token URL	OAuth2TokenServiceURL	Examples: <ul style="list-style-type: none"> <li>• US: <a href="https://api.ariba.com/v2/oauth/token">https://api.ariba.com/v2/oauth/token</a></li> <li>• Europe: <a href="https://api-eu.ariba.com/v2/oauth/token">https://api-eu.ariba.com/v2/oauth/token</a></li> <li>• UAE: <a href="https://api.mn1.ariba.com/v2/oauth/token">https://api.mn1.ariba.com/v2/oauth/token</a></li> </ul>
OAuth Client ID	User	Alphanumeric string Example: <b>aaaa12345-1111-3333-cccc-1234567890</b>
OAuth Secret	Password	Alphanumeric string Example: <b>aaaGGG1eee12abcdefGHIJK123lmnopTTT</b>
Application key	<code>ariba.applications.api.key</code>	Alphanumeric string Example: <b>123abc123XYZ000abc123ABC012345</b>
AN-ID	<code>ariba.applications.realm.id</code>	AN<numeric_string> Example: <b>AN000111222333</b>

## Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

After fulfilling the prerequisites, you can create an SAP Ariba Applications proxy system to load its users into an on-premise system and provision groups and new users back to SAP Ariba Applications.

These proxy systems consume SCIM 2.0 API provided by SAP Ariba Applications. For more information about the SAP Ariba SCIM API scope of support, see [3228340](#).

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#) standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.emails[0].value",
  "targetPath": "$.emails[0].value",
  "optional": true
},
```

You also set the following filter in the [Properties](#) tab: `ariba.applications.user.filter = addresses.country eq "US"`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=emails[0].value eq "john.smith03@dummymail.com"`

The query request to the SAP Ariba Applications API will result into: `/Users?filter=addresses.country eq "US" and emails[0].value eq "john.smith03@dummymail.com"`

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

#### SAP Cloud Identity Infrastructure

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

#### Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*



## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Ariba Applications](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>

Property Name	Description & Value
URL	Enter the SCIM API URL for your SAP Ariba application (see the <b>Prerequisites</b> section).
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth Client ID (see the <b>Prerequisites</b> section).
Password	Enter the OAuth Secret (see the <b>Prerequisites</b> section).
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL (see the <b>Prerequisites</b> section).
ariba.applications.api.key	Enter your application key (see the <b>Prerequisites</b> section).
ariba.applications.realm.id	Enter your AN-ID (see the <b>Prerequisites</b> section).
(Optional) ariba.applications.group.flatten	<p>This property allows or forbids reading "nested groups" (group structures) from SAP Ariba Applications. If enabled (<b>true</b>), group members of type <i>group</i> will be ignored during read in order not to be provisioned to target systems that do not support nested groups. Thus, leave the default/predefined <b>false</b> value only if you are sure that the consuming external application (identity management system) supports nested groups.</p> <p>Default value: <i>false</i></p> <p>Predefined value (during system creation): <i>false</i></p>
(Optional) ariba.applications.support.patch.operation	Default value: <i>true</i>

Exemplary destination (property configuration):

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://openapi.ariba.com
User=aaaa12345-1111-3333-cccc-1234567890
Password=*****
OAuth2TokenServiceURL=https://api.ariba.com/v2/oauth/token
ariba.applications.group.flatten=false
ariba.applications.support.patch.operation=true
ariba.applications.api.key=123abc123XYZ000abc123ABC012345
ariba.applications.realm.id=AN000111222333
```

---

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Ariba Applications* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Ariba Applications. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Ariba APIs Portal](#) ➔ *Discover* ➔ *SUPPLIER MANAGEMENT*

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users* or */Groups*) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.schemas"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.emails[0].value",
        "targetPath":
"$$.emails[0].value",
        "optional": true
      },
      {
        "sourcePath":
"$$.emails[?(@.primary==
true)].value",
        "optional": true,

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "sourcePath":
"$$.Operations",
        "targetPath":
"$$.Operations",
        "preserveArrayWithSingleElement":
true,
        "scope":
"patchEntity"
      },
      {
        "sourcePath":
"$$.schemas",
        "targetPath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "scope":
"patchEntity"
      },
      {
        "targetPath":
"$$.id",
        "type": "remove",
        "scope":
"patchEntity"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$$.schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath":
"$$.schemas[1]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:sap:2.0:User",
        "targetPath":
"$$.schemas[2]"

```

```

"correlationAttribute": true
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ]
[ 'userUuid' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ]
[ 'userUuid' ]",
        "optional": true
    },
    {
        "sourcePath":
"$$.displayName",
        "optional": true,
        "targetPath":
"$$.displayName"
    },
    {
        "sourcePath":
"$$.active",
        "optional": true,
        "targetPath":
"$$.active"
    },
    {
        "sourcePath":
"$$.title",
        "optional": true,
        "targetPath":
"$$.title"
    },
    {
        "sourcePath":
"$$.locale",
        "optional": true,
        "targetPath":
"$$.locale",
        "functions": [
            {
                "type":
"substring",
                "beginIndex": 0,
                "endIndex": 2
            }
        ]
    },
    {
        "sourcePath":
"$$.timezone",
        "optional": true,
        "targetPath":
"$$.timezone"
    },
    {
        "sourcePath":
"$$.phoneNumbers",

```

```

    },
    {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName"
    },
    {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName",
        "optional": true
    },
    {
        "sourcePath":
"$$.emails",
        "targetPath":
"$$.emails",
        "preserveArrayWithSingleElement":
true,
        "optional": true
    },
    {
        "condition":
"$$.emails[0].length() > 0",
        "constant": true,
        "targetPath":
"$$.emails[0].primary"
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ]
[ 'userUuid' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ] [ 'userUuid' ]"
    },
    {
        "sourcePath":
"$$.locale",
        "optional": true,
        "targetPath":
"$$.locale"
    },
    {
        "sourcePath":
"$$.active",
        "targetPath":
"$$.active"
    },
    {
        "sourcePath":
"$$.timezone",
        "optional": true,
        "targetPath":
"$$.timezone"
    },
    {

```

```

"preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath":
"$ .phoneNumbers"
    },
    "sourcePath":
"$ .groups",
"preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath":
"$ .groups"
    },
    "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
    "optional": true
    }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "sourcePath":
"$ .id",
                "targetPath":
"$ .id",
                "targetVariable":
"entityIdSourceSystem"
            },
            {
                "sourceVariable":
"entityBaseLocation",
                "targetVariable":
"entityLocationSourceSystem",
                "targetPath":
"$ .meta.location",
                "functions": [
                    {
                        "type":
"concatString",
                        "suffix":
"${entityIdSourceSystem}"
                    }
                ]
            }
        ],
        "sourcePath":
"$ .schemas",

```

```

    "sourcePath":
"$ .phoneNumbers",
"preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath":
"$ .phoneNumbers",
    "functions": [
        {
            "function": "putIfAbsent",
            "key":
"type",
            "defaultValue": "work"
        }
    ],
    "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
    "optional": true
    }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "sourceVariable":
"entityIdTargetSystem",
                "targetPath":
"$ .id"
            },
            {
                "sourcePath":
"$ .Operations",
                "targetPath":
"$ .Operations",
                "preserveArrayWithSingleElement":
true,
                "scope":
"patchEntity"
            },
            {
                "sourcePath":
"$ .schemas",
                "targetPath":
"$ .schemas",
                "preserveArrayWithSingleElement":
true,
                "scope":
"patchEntity"
            }
        ]
    }

```

## Read Transformation

```
"preserveArrayWithSingleElement":
true,
    "targetPath":
    "$.schemas"
    },
    {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.displayName"
    },
    {
        "sourcePath":
        "$.members",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetPath":
        "$.members"
    }
    ]
}
```

## Write Transformation

```
    },
    {
        "targetPath":
        "$.id",
        "type": "remove",
        "scope":
        "patchEntity"
    },
    {
        "constant":
        "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath":
        "$.schemas[0]"
    },
    {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.displayName"
    },
    {
        "sourcePath":
        "$.members",
        "preserveArrayWithSingleElement":
        true,
        "targetPath":
        "$.members",
        "optional": true
    }
    ]
}
```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

### i Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[The SAP Ariba developer portal](#)

[Video: Create application and API approval process](#) 📺

## 1.6.3.8 SAP BTP ABAP environment

Follow this procedure to set up SAP BTP ABAP environment as a proxy system.

### Prerequisites

- You have user credentials for an external back-end system with read and write permissions.
- To establish the connection between Identity Provisioning and SAP BTP ABAP environment, you need to set up the communication (user, system and arrangement) on SAP BTP ABAP environment. You can do it now (as a prerequisite) or in the process of configuring SAP BTP ABAP environment as a proxy system, as described in step 5.

### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).


## Context

You can use SAP BTP ABAP environment as a proxy connector to execute *hybrid* scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to SAP BTP ABAP environment, whenever the external back-end requests such. This scenario supports:

- Reading of **business users** (Employee) and **business roles** (which are considered as *groups*)
- Writing of **users** and **assignments**

### SCIM Filtering Support



The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names ([<schema>:<attribute>](#)) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

#### ❖ Example

```
{
  "sourcePath": "$.user.userName",
  "targetPath": "$.userName",
  "optional": true,
  "correlationAttribute": true
},
```

Since SAP BTP ABAP environment doesn't support user filtering, then:

If, for example, the SCIM Proxy endpoint request is: **GET /Users?filter=username eq "JOHNSMITH003"**

The query request to the SAP BTP ABAP environment API will result into a search for a user whose username is 'JOHNSMITH003'.

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

## i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.


2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <a href="#">SAP Cloud Identity Infrastructure</a>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <a href="#">SAP BTP, Neo Environment</a>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li><li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li><li>3. Save your changes.</li><li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li><li>5. Save your changes.</li></ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li><li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li><li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li><li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li><li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li><li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li><li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li><li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li></ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP BTP ABAP environment](#) as a proxy system. For more information, see: [Add a System \[page 1477\]](#).
5. Set up the communication between Identity Provisioning and SAP BTP ABAP environment and configure your authentication method (basic or certificate-based).

## i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP BTP ABAP environment proxy system, select the [Certificate](#) tab and choose [Generate](#) > [Download](#) , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP BTP ABAP environment backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide [User Name](#) and [Password](#).

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide [System ID](#), [System Name](#) and [Host Name](#).

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose [Scenario ID](#) SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

For more information, see [Maintain a Communication Arrangement for Inbound Communication](#) .

#### **i** Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

6. Choose the [Properties](#) tab to configure the connection settings for your system.

#### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.



If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>

Property Name	Description & Value
URL	<p>Specify the API URL to your SAP BTP ABAP environment system.</p> <p>You can take the URL from the communication scenario SAP_COM_0193.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"> <li>• <a href="#">BasicAuthentication</a></li> <li>• <a href="#">ClientCertificateAuthentication</a></li> </ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div> <b>! Restriction</b>  Do not use special symbol ',' (comma) as it is not supported. </div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b>  Do not use special symbol ',' (comma) as it is not supported. </div>
a4c.skip.read.archived	<p>In the event of archived (disabled) entities in your SAP BTP ABAP environment system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
ips.date.variable.format	<p><a href="#">yyyy-MM-dd</a></p> <p>(needed for the <a href="#">Read Transformation</a>)</p>

Property Name	Description & Value
<code>a4c.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP BTP ABAP environment proxy system in a hybrid scenario.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p>
(Optional) <code>a4c.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP BTP ABAP environment system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a>  → <b>4.5 Filter System Query Option</b></p>

Property Name	Description & Value
<code>a4c.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP BTP ABAP environment target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>personExternalID</code></li> <li><code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p>
(Optional) <code>a4c.roles.page.size</code>	<p>Indicate how many business roles (considered as <code>groups</code>) per page to be read from your SAP BTP ABAP environment system.</p> <p>The value must be an integer number.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://12345-aaaaa-3333.abap.hana.ondemand.com
User=MyABAPEnvUser
Password=*****
ips.date.variable.format=yyyy-MM-dd
a4c.skip.read.archived=true
a4c.user.roles.override=false
a4c.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true
a4c.roles.page.size=30
```

---

7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP BTP ABAP environment](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP BTP ABAP environment system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

Default read and write transformations:

→ Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy [Read Transformation](#) is used for [write](#) cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$$.personID",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ],
        {
          "sourcePath":
"$$.personalInformation.firstName",
          "targetPath":
"$$.name.givenName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.lastName",
          "targetPath":
"$$.name.familyName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.middleName",
          "targetPath":
"$$.name.middleName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.personFullName",
          "targetPath":
"$$.name.formatted",
          "optional": true
        },

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
        "optional": true,
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
[ 'userUuid' ]",
        "optional": true,
        "targetPath":
"$$.user.globalUserID"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.personID"
      },
      {
        "targetPath":
"$$.businessPartnerRoleCode",
        "type": "valueMapping",
        "sourcePaths": [
          "$.userType"
        ],
        "defaultValue": "BUP003",
        "valueMappings": [
          {
            "key": [
              "Employee"
            ],
            "mappedValue":
"BUP003"
          }
        ],
        "scope": "createEntity",
        "sourceVariable":
"currentDate",
        "targetPath":
"$$.validityPeriod.startDate"
      },
      {
        "scope": "createEntity",
        "constant": "9999-12-31",

```



```

      "sourcePath":
"$ .user.userName",
      "targetPath":
"$ .userName",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "constant": true,
      "targetPath": "$ .active"
    },
    {
      "condition":
"$ .user.lockedIndicator ==
'true'",
      "constant": false,
      "targetPath": "$ .active",
      "optional": true
    },
    {
      "condition":
"($ .user.validityPeriod.startDate
> '{currentDate}') ||
('{currentDate}' >
$.user.validityPeriod.endDate)",
      "constant": false,
      "optional": true,
      "targetPath": "$ .active"
    },
    {
      "sourcePath":
"$ .workplaceInformation.emailAddress",
      "targetPath":
"$ .emails[0].value",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .user.logonLanguageCode",
      "optional": true,
      "targetPath": "$ .locale"
    },
    {
      "sourcePath":
"$ .PersonExternalID",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .user.role[*].roleName",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath": "$ .groups[?
(@.value)]"

```

```

      "targetPath":
"$ .validityPeriod.endDate"
    },
    {
      "scope": "createEntity",
      "sourceVariable":
"currentDate",
      "targetPath":
"$ .user.validityPeriod.startDate"
    },
    {
      "scope": "createEntity",
      "constant": "9999-12-31",
      "targetPath":
"$ .user.validityPeriod.endDate"
    },
    {
      "sourcePath":
"$ .name.givenName",
      "optional": true,
      "targetPath":
"$ .personalInformation.firstName"
    },
    {
      "sourcePath":
"$ .name.familyName",
      "targetPath":
"$ .personalInformation.lastName"
    },
    {
      "sourcePath":
"$ .name.middleName",
      "optional": true,
      "targetPath":
"$ .personalInformation.middleName"
    },
    {
      "sourcePath":
"$ .name.formatted",
      "optional": true,
      "targetPath":
"$ .personalInformation.personFullName"
    },
    {
      "sourcePath":
"$ .userName",
      "targetPath":
"$ .user.userName"
    },
    {
      "sourcePath":
"$ .nickName",
      "optional": true,
      "targetPath":
"$ .user.nickName"
    },
    {
      "sourcePath": "$ .locale",
      "optional": true,
      "targetPath":
"$ .user.logonLanguageCode"
    },

```

```

    },
    {
      "sourcePath":
"$$.user.globalUserID",
      "optional": true,
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ][ 'userUuid' ]"
    },
    {
      "type": "valueMapping",
      "sourcePaths": [
        "$$.user.timeZoneCode"
      ],
      "targetPath":
"$$.timezone",
      "defaultValue": "Europe/
Berlin",
      "valueMappings": [
        {
          "key": [
            "WDFT"
          ],
          "mappedValue":
"Europe/Berlin"
        },
        {
          "key": [
            "ISRAEL"
          ],
          "mappedValue": "Asia/
Jerusalem"
        },
        {
          "key": [
            "RUS03"
          ],
          "mappedValue":
"Europe/Moscow"
        },
        {
          "key": [
            "AUSNSW"
          ],
          "mappedValue":
"Australia/Sydney"
        },
        {
          "key": [
            "UTC+4"
          ],
          "mappedValue": "Asia/
Dubai"
        },
        {
          "key": [
            "BRAZIL"
          ],
          "mappedValue":
"America/Sao_Paulo"
        }
      ]
    }
  ],
  "scimEntityEndpoint": "Users"
},
{
  "group": {
    "mappings": [
      {
        "sourcePath":
"$$.displayName",
        "targetVariable":
"entityIdTargetSystem",
        "scope": "createEntity"
      },
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
      },
      {
        "sourcePath":
"$$.members[*].value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$$.members[?
(@.value)]"
      },
      {
        "sourcePath":
"$$.Operations",
        "targetPath":
"$$.Operations",
        "preserveArrayWithSingleElement":
true,
        "scope": "patchEntity"
      },
      {
        "sourcePath": "$$.schemas",
        "targetPath": "$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "scope": "patchEntity"
      }
    ]
  },
  "scimEntityEndpoint": "Groups"
}

```

```

    {
      "sourcePath":
"$$.emails[0].value",
      "optional": true,
      "targetPath":
"$$.workplaceInformation.emailAddre
ss"
    },
    {
      "condition": "$$.active ==
false",
      "constant": "true",
      "targetPath":
"$$.user.lockedIndicator"
    },
    {
      "scimEntityEndpoint": "Users"
    },
    {
      "group": {
        "mappings": [
          {
            "sourcePath":
"$$.displayName",
            "targetVariable":
"entityIdTargetSystem",
            "scope": "createEntity"
          },
          {
            "sourcePath":
"$$.displayName",
            "targetPath":
"$$.displayName"
          },
          {
            "sourcePath":
"$$.members[*].value",
            "preserveArrayWithSingleElement":
true,
            "optional": true,
            "targetPath": "$$.members[?
(@.value)]"
          },
          {
            "sourcePath":
"$$.Operations",
            "targetPath":
"$$.Operations",
            "preserveArrayWithSingleElement":
true,
            "scope": "patchEntity"
          },
          {
            "sourcePath": "$$.schemas",
            "targetPath": "$$.schemas",
            "preserveArrayWithSingleElement":
true,
            "scope": "patchEntity"
          }
        ]
      },
      "scimEntityEndpoint": "Groups"
    }
  ],
  "scimEntityEndpoint": "Groups"
}

```

```

        "key": [
            "BRZLEA"
        ],
        "mappedValue":
"America/Sao_Paulo"
    },
    {
        "key": [
            "MSTNO"
        ],
        "mappedValue":
"America/Phoenix"
    },
    {
        "key": [
            "EST"
        ],
        "mappedValue":
"America/New_York"
    },
    {
        "key": [
            "UTC"
        ],
        "mappedValue": "Etc/
UTC"
    },
    {
        "key": [
            "UTC+3"
        ],
        "mappedValue": "Asia/
Riyadh"
    },
    {
        "key": [
            "EST_"
        ],
        "mappedValue":
"America/Toronto"
    },
    {
        "key": [
            "UTC+8"
        ],
        "mappedValue": "Asia/
Shanghai"
    },
    {
        "key": [
            "JAPAN"
        ],
        "mappedValue": "Asia/
Tokyo"
    }
]
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.businessPartnerRoleCode"
    ]
}

```

```

    }
}

```

```

    ],
    "targetPath":
"$ .userType",
    "defaultValue":
"Employee",
    "valueMappings": [
        {
            "key": [
                "BUP003"
            ],
            "mappedValue":
"Employee"
        }
    ]
},
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "sourcePath": "$ .ID",
                "targetPath": "$ .id",
                "targetVariable":
"entityIdSourceSystem"
            },
            {
                "sourceVariable":
"entityBaseLocation",
                "targetVariable":
"entityLocationSourceSystem",
                "targetPath":
"$ .meta.location",
                "functions": [
                    {
                        "type":
"concatString",
                        "suffix": "$
{entityIdSourceSystem}"
                    }
                ]
            },
            {
                "sourcePath": "$ .ID",
                "targetPath":
"$ .displayName"
            },
            {
                "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
                "targetPath":
"$ .schemas[0]"
            },
            {
                "sourcePath":
"$ .to_BusinessUserAssignment.resul
ts",
                "optional": true,

```

```

"preserveArrayWithSingleElement":
true,
    {
        "targetPath": "$.members"
    },
    {
        "type": "remove",
        "targetPath":
"$$.members[*].__metadata"
    },
    {
        "type": "remove",
        "targetPath":
"$$.members[*].UserName"
    },
    {
        "type": "rename",
        "constant": "value",
        "targetPath":
"$$.members[*].PersonID"
    },
    {
        "constant": "User",
        "targetPath":
"$$.members[*].type"
    }
]
}
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource in the [Read Transformation](#). To learn how, see [Guided Answers: Business Role Description](#).

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

### **i** Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### **⚠** Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.9 SAP BTP Account Members (Neo)

Follow this procedure to set up your SAP Business Technology Platform as a proxy system, from and to which you can provision and write members in your Neo account.

### Prerequisites

- You have created a new Platform API OAuth client, with API [Account Member Management](#) and scopes [Manage Account Members](#) and [Read Account Members](#). Save the [Client ID](#) and [Client Secret](#) as you'll need them when you configure your proxy system. Make sure you save the client secret as you cannot retrieve it later. For more information, see [Create a Platform API Client](#).

#### Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).


### Context

The Identity Provisioning service helps companies to automatically manage the user-to-platform roles assignments for SAP Business Technology Platform subaccounts.

You can use SAP Business Technology Platform as a proxy connector to execute [hybrid](#) scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to SAP Business Technology Platform, whenever the external back-end requests such.

The Identity Provisioning service system SAP Business Technology Platform Account Members represents the various authorization mappings managed by the Platform Authorization Management API of SAP BTP as one generic SCIM 2.0 system, with some limitations. For more information, see: [Platform Authorization Management API](#).

#### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- 0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- 1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example:  
`GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber eq '<attribute>'`
- If your system supports multivalued e-mails (that is `$.emails[0].value`, `$.emails[1].value`, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (`$.emails[0].value`).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.emails[0].value",
  "targetPath": "$.emails[0].value",
  "optional": true
},
```

You also set the following filter in the [Properties](#) tab: `scim.user.filter = name.familyName eq "Smith"`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=emails[0].value eq "john.smith03@dummymail.com"`

The query request to the SAP BTP API will result into: `/Users?filter=name.familyName eq "Smith" and emails[0].value eq "john.smith03@dummymail.com"`

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.



## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP BTP Account Members (Neo)* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter: <a href="#">https://api.&lt;SAP_BTP_host&gt;/authorization/v1/platform/accounts/&lt;SAP_BTP_account&gt;</a> Examples: <ul style="list-style-type: none"> <li>(Europe – Rot) <a href="#">https://api.hana.ondemand.com/authorization/v1/platform/accounts/abc123xyz</a></li> <li>(Japan – Tokyo) <a href="#">https://api.jp1.hana.ondemand.com/authorization/v1/platform/accounts/abc123xyz</a></li> </ul>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the Client ID of the new Platform API OAuth client created for the <b>Account Member Management</b> API (see the prerequisites).
Password	Enter the Client Secret of the new Platform API OAuth client created for the <b>Account Member Management</b> API (see the prerequisites).
OAuth2TokenServiceURL	Enter: <a href="#">https://api.&lt;SAP_BTP_host&gt;/oauth2/apitoken/v1</a> Examples: <ul style="list-style-type: none"> <li>(Europe – Rot) <a href="#">https://api.hana.ondemand.com/oauth2/apitoken/v1</a></li> <li>(US East – Sterling) <a href="#">https://api.us3.hana.ondemand.com/oauth2/apitoken/v1</a></li> </ul>
scp.user.userbase	This property specifies the host to the identity provider to be used with this proxy system. All provisioned users can be authenticated only by this identity provider. Default value: <a href="#">account.sap.com</a> If you use another IdP, enter its value as configured in the SAP BTP cockpit. For example: <a href="#">&lt;account_ID&gt;.accounts.ondemand.com</a>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP BTP Account Members](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your proxy system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP BTP: Authorization Management API](#)

The *SAP BTP Account Members (Neo)* proxy connector supports SCIM PATCH operation.

The read transformation takes the group name from the source system. However, the names of the SAP BTP platform roles sometimes contain characters that do not fit the SCIM URI specification. As Identity Provisioning service cannot directly use these characters as a SCIM resource ID, it needs to first encode them into base32 ASCII format. The intermediate JSON logic of the proxy system will then use the new encoded ID and write it to the target system. That means, the ID will be decoded in the proxy write transformation during the **patchEntity** operation. For more information, see: [Transformation Expressions \[page 330\]](#) → *patchEntity*

Default read and write transformations:

#### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta .location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .displayName"
      },
      {
        "sourcePath":
"$ .name .givenName",
        "optional": true,
        "targetPath":
"$ .name .givenName"
      },
      {
        "sourcePath":
"$ .name .familyName",
        "optional": true,
        "targetPath":
"$ .name .familyName"
      },
      {
        "sourcePath":
"$ .emails[0].value",
        "optional": true,

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath": "$ .userName"
      },
      {
        "sourcePath":
"$ .groups[*].value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$ .roles[?
(@.value)]"
      },
      {
        "constant":
"%scp.user.userbase%",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:UserExt']
['userbase']"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "constant":
"urn:sap:cloud:scim:schemas:extension:custom:2.0:UserExt",
        "targetPath":
"$ .schemas[1]"
      }
    ],
    "scimEntityEndpoint": "Users"
  },
  "group": {
    "mappings": [
      {
        "sourcePath":
"$ .Operations",
        "targetPath":
"$ .Operations",
        "preserveArrayWithSingleElement":
true,
        "scope": "patchEntity"

```

```

        "targetPath":
"$ .emails[0].value"
      },
      {
        "sourcePath":
"$ .description",
        "optional": true,
        "targetPath":
"$ .description"
      },
      {
        "sourcePath": "$.schemas",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath":
"$ .roles[*].value",

"preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$.groups[?
(@.value)]"
      }
    ],
    "group": {
      "scimEntityEndpoint":
"Groups",
      "mappings": [
        {
          "sourcePath": "$ .id",
          "targetPath": "$ .id",
          "targetVariable":
"entityIdSourceSystem"
        },
        {
          "sourceVariable":
"entityBaseLocation",
          "targetVariable":
"entityLocationSourceSystem",
          "targetPath":
"$ .meta.location",
          "functions": [
            {
              "type":
"concatString",
              "suffix": "$
{entityIdSourceSystem}"
            }
          ]
        }
      ],
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName"
      },
      {

```

```

    },
    {
      "sourcePath": "$.schemas",
      "targetPath": "$.schemas",

"preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    }
  ],
  "scimEntityEndpoint": "Groups"
}

```

```

        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
        "targetPath":
"$$.schemas[0]"
    },
    {
        "sourcePath":
"$$.members[*].value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$$.members[?
(@.value)]"
    },
    {
        "constant": "User",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.members[*].type"
    }
]
}
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

#### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

### 1.6.3.10 SAP BTP Java/HTML5 apps (Neo)

Follow this procedure to set up SAP Business Technology Platform as a proxy system, from and to which you can provision user-to-groups assignments for Java/HTML5 applications that run on SAP BTP, Neo environment.

## Prerequisites

- You have created a new Platform API OAuth client, with [Authorization Management](#) scopes. Save the [Client ID](#) and [Client Secret](#) as you'll need them when you configure your proxy system. Make sure you save the client secret as you cannot retrieve it later.  
To learn how to create a Platform API client and what are the required platform service roles, see: [Create a Platform API Client](#)

## i Note


Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

The Identity Provisioning service helps companies to automatically manage the user-to-groups assignments for Java/HTML5 applications running on the SAP Business Technology Platform. For this aim, the service reuses data from an active company user store.

For this scenario, SAP Business Technology Platform is a proxy system. That is, the Identity Provisioning service represents the various authorization mappings managed by Authorization Management REST API of SAP BTP as one generic SCIM 2.0 system, with some limitations. For more information, see: [Using the Authorization Management REST API](#)

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
```



```
    "correlationAttribute": true  
  },
```

If, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP BTP API will result into: **/Users?filter=userName eq "johnsmith03"**

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

#### SAP Cloud Identity Infrastructure

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

#### Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP BTP Java/HTML5 apps \(Neo\)](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>

Property Name	Description & Value
URL	<p>Enter: <b><code>https://api.&lt;SAP_BTP_host&gt;/authorization/v1/accounts/&lt;SAP_BTP_account&gt;</code></b></p> <p>Examples:</p> <ul style="list-style-type: none"> <li>(Europe – Rot) <a href="https://api.hana.ondemand.com/authorization/v1/accounts/abc123xyz">https://api.hana.ondemand.com/authorization/v1/accounts/abc123xyz</a></li> <li>(Japan – Tokyo) <a href="https://api.jp1.hana.ondemand.com/authorization/v1/accounts/abc123xyz">https://api.jp1.hana.ondemand.com/authorization/v1/accounts/abc123xyz</a></li> </ul>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the Client ID of the new Platform API OAuth client created for the Authorization Management API (see the prerequisites).
Password	Enter the Client Secret of the new Platform API OAuth client created for the Authorization Management API (see the prerequisites).
OAuth2TokenServiceURL	<p>Enter: <b><code>https://api.&lt;SAP_BTP_host&gt;/oauth2/apitoken/v1</code></b></p> <p>Examples:</p> <ul style="list-style-type: none"> <li>(Europe – Rot) <a href="https://api.hana.ondemand.com/oauth2/apitoken/v1">https://api.hana.ondemand.com/oauth2/apitoken/v1</a></li> <li>(US East – Sterling) <a href="https://api.us3.hana.ondemand.com/oauth2/apitoken/v1">https://api.us3.hana.ondemand.com/oauth2/apitoken/v1</a></li> </ul>
(Optional) <code>hcp.read.group.roles</code>	<p>If you set this property to <i>true</i>, the Identity Provisioning will read the following additional attributes for a SAP BTP group:</p> <ul style="list-style-type: none"> <li>Application roles</li> <li>Group mappings, defined by your identity provider</li> </ul> <div> <p><b>! Restriction</b></p> <p>This property and the relevant functionality is applicable only if SAP BTP and the external SCIM-based system belong to one and the same region.</p> </div>
(Optional) <code>hcp.patch.response.with.resource</code>	<p>Use this property when you execute hybrid scenarios with SAP BTP as a SCIM proxy system, and you update an entity (mostly relevant to groups, like when you change the members of a group) via a PATCH request.</p> <p>Set this property to <i>true</i>. This way, the successful PATCH request will return a response code 200 (<i>OK</i>) back to the consumer client application with a payload body containing the updated attributes of the relevant group.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP BTP Java/HTML5 apps (Neo)* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your proxy system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP BTP: Authorization Management API](#)

The *SAP BTP Java/HTML5 apps (Neo)* proxy connector supports SCIM PATCH operation.

The read transformation takes the group name from the source system. However, SAP BTP groups names sometimes contain characters that do not fit the SCIM URI specification. As Identity Provisioning service cannot directly use these characters as a SCIM resource ID, it needs to first encode them into base32 ASCII format. The intermediate JSON logic of the proxy system will then use the new encoded ID and write it to the target system. That means, the ID will be decoded in the proxy write transformation during the **patchEntity** operation. For more information, see: [Transformation Expressions \[page 330\]](#) → *patchEntity*

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## ≡, Code Syntax

```
{
  "group": {
    "scimEntityTypeEndpoint":
"Groups",
    "mappings": [
      {
        "sourcePath": "$.name",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem",
        "functions": [
          {
            "type": "encode",
            "algorithm": "base32",
            "skipPadding": true
          }
        ]
      },
      {
        "targetPath":
"$$.displayName",
        "sourcePath": "$.name"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "targetPath":
"$$.schemas[0]",
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
        "sourcePath": "$.members",
        "preserveArrayWithSingleElement":
true,
        "targetPath": "$.members[?
(@.value)]",
        "optional": true
      },
      {
        "sourcePath": "$.roles",
        "targetPath": "$.roles",

```

## Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$.userName",
        "targetPath": "$.name",
        "targetVariable":
"entityIdTargetSystem"
      }
    ]
  },
  "group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
      {
        "sourcePath":
"$.displayName",
        "targetPath": "$.name",
        "targetVariable":
"entityIdTargetSystem"
      },
      {
        "scope": "deleteEntity",
        "sourceVariable":
"entityIdTargetSystem",
        "targetVariable":
"entityIdTargetSystem",
        "functions": [
          {
            "type": "decode",
            "algorithm": "base32",
            "skipPadding": true
          },
          {
            "type": "toString"
          }
        ]
      },
      {
        "scope": "patchEntity",
        "sourceVariable":
"entityIdTargetSystem",
        "targetVariable":
"entityIdTargetSystem",
        "functions": [
          {
            "type": "decode",
            "algorithm": "base32",
            "skipPadding": true
          },
          {
            "type": "toString"
          }
        ]
      }
    ]
  }
},
```

```

"preserveArrayWithSingleElement":
true,
    "optional": true
    },
    {
        "sourcePath":
"$ .idpGroupMappings",
        "targetPath":
"$ .idpGroupMappings",

"preserveArrayWithSingleElement":
true,
    "optional": true
    },
    {
        "constant": "User",

"preserveArrayWithSingleElement":
true,
        "targetPath":
"$ .members[*].type",
        "optional": true
    }
    ],
    },
    "user": {
        "scimEntityEndpoint": "Users",
        "mappings": [
            {
                "sourcePath": "$ .name",
                "targetPath": "$ .id",
                "targetVariable":
"entityIdSourceSystem"
            },
            {
                "targetPath":
"$ .displayName",
                "sourcePath": "$ .name"
            },
            {
                "targetPath":
"$ .userName",
                "sourcePath": "$ .name",
                "correlationAttribute":
true
            },
            {
                "sourceVariable":
"entityBaseLocation",
                "targetVariable":
"entityLocationSourceSystem",
                "targetPath":
"$ .meta.location",
                "functions": [
                    {
                        "type":
"concatString",
                        "suffix": "$
{entityIdSourceSystem}"
                    }
                ]
            }
        ]
    }

```

```

        "sourcePath":
"$ .Operations",
        "targetPath":
"$ .Operations",

"preserveArrayWithSingleElement":
true,
    "scope": "patchEntity"
    },
    {
        "sourcePath": "$ .schemas",
        "targetPath": "$ .schemas",

"preserveArrayWithSingleElement":
true,
    "scope": "patchEntity"
    },
    {
        "sourcePath":
"$ .members[*].value",
        "optional": true,
        "targetPath": "$ .users"
    }
    ]
}

```

```

    },
    {
      "targetPath":
"$ .schemas[0]",
      "constant":
"urn:ietf:params:scim:schemas:core:2.0:User"
    },
    {
      "sourcePath": "$ .groups",

      "preserveArrayWithSingleElement":
true,
      "targetPath": "$ .groups[?
(@.value)]",
      "optional": true,
      "functions": [
        {
          "type": "encode",
          "algorithm": "base32",
          "skipPadding": true
        }
      ]
    },
    {
      "constant": "direct",

      "preserveArrayWithSingleElement":
true,
      "targetPath":
"$ .groups[*].type",
      "optional": true
    }
  ]
}
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

#### Neo Environment

- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>• For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul> | <ul style="list-style-type: none"> <li>• For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul> |
|---|--|

### **i Note**

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## **Next Steps**

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### **⚠ Caution**

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## **Related Information**

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)



## 1.6.3.11 SAP BTP XS Advanced UAA (Cloud Foundry)

Follow this procedure to set up the SAP BTP XS Advanced UAA (running on SAP BTP, Cloud Foundry environment) as a proxy system.

### Prerequisites

- You have a technical database user with administrator permissions for SAP BTP XS Advanced UAA to read, create and update user account information. To learn how, see: [Get API Access](#)
- Since OAuth is used for authentication of your service instance, you need to generate a service key for the service instance, and then retrieve this service key with OAuth 2.0 client credentials (client ID and secret). You'll use them when creating a destination (or specifying the Identity Provisioning connection properties) for access token retrieval. To learn how to generate XSUAA OAuth credentials, see: [Retrieve Credentials for Remote Applications](#)

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

In simple terms, XS Advanced is basically the Cloud Foundry open-source PaaS with a number of tweaks and extensions provided by SAP. These SAP enhancements include integration with the SAP HANA database, OData support, compatibility with XS classic model, and some additional features designed to improve application security. XS Advanced also provides support for business applications that are composed of multiple micro-services, also known as multi-target applications.


SAP BTP XS Advanced UAA is responsible for the connection of identity providers with business users (for applications). SAP BTP XS Advanced UAA provides authorizations on application level: [role collections](#), [roles](#), [attributes](#), and [role templates](#). To learn more, see: [What Is the SAP Authorization and Trust Management Service?](#)

You can use SAP BTP XS Advanced UAA as a proxy connector to execute [hybrid](#) scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to SAP BTP XS Advanced UAA, whenever the external back-end requests such.

#### → Remember

You can manage users and groups to SAP BTP XS Advanced UAA on **application** level only. You cannot manage them on a [subaccount](#) level.

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to

users. If your system supports *native read filtering*, the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the *Read Transformation*, there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
*GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the *Properties* tab: *scim.user.filter = timezone eq "US"*

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the API of SAP BTP XS Advanced UAA will result into: **/Users?filter=timezone eq "US" and userName eq "johnsmith03"**

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

## Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li><li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li><li>3. Save your changes.</li><li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li><li>5. Save your changes.</li></ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li><li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li><li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li><li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li><li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li><li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li><li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li><li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li></ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP BTP XS Advanced UAA (Cloud Foundry)* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL to the SCIM API of your SAP BTP XS Advanced UAA system.</p> <p>If not sure about the exact URL, ask your SAP BTP XS UAA administrator.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
OAuth2TokenServiceURL	<p>As you need to make OAuth authentication to the SAP BTP XS Advanced UAA system, enter the URL to the OAuth2 token service</p> <p>If not sure about the exact URL, ask your SAP BTP XS UAA administrator.</p>
User	Enter the OAuth client ID of the SAP BTP XS Advanced UAA technical user (see <b>Prerequisites</b> ).
Password	(Credential) Enter the OAuth client secret of the technical user (see <b>Prerequisites</b> ).
xsuaa.origin	<p>Enter the location of your identity provider. To do this:</p> <ol style="list-style-type: none"><li>1. Open your SAP BTP cockpit.</li><li>2. Go to your Cloud Foundry global account and choose your subaccount.</li><li>3. From the left-side navigation, choose <a href="#">Trust Configuration</a>.</li><li>4. Copy/paste the <a href="#">Origin Key</a> value.</li></ol> <p>This value will be used as the <a href="#">origin</a> attribute in the system transformation.</p> <p>For more information, see <a href="#">Configure Single and Multiple Origins [page 1140]</a></p>

Property Name	Description & Value
<code>xsuaa.origin.filter.enabled</code>	<p>This flag property depends on <code>xsuaa.origin</code>. Possible values: <b>true</b> or <b>false</b></p> <ul style="list-style-type: none"> <li>If set to <b>true</b>, the Identity Provisioning service will read only users whose identity provider is set as a value of <code>xsuaa.origin</code>.</li> <li>If set to <b>false</b>, the Identity Provisioning service will read all users, regardless of their origin.</li> <li>If set to <b>true</b> but the <code>xsuaa.origin</code> property is missing, the provisioning will fail.</li> </ul>
<code>scim.support.patch.operation</code>	<p>Use this property if you want to modify the members of a group.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li><b>true</b> – the Identity Provisioning service will modify the group membership via the <a href="#">PATCH /Groups</a> endpoint of UAA. To learn how, see <a href="#">Patch</a>.</li> <li><b>false</b> – the Identity Provisioning service will modify the group membership via the <a href="#">POST /Groups</a> or <a href="#">DELETE /Groups</a> endpoints of UAA. To learn how, see <a href="#">Add Member</a> and <a href="#">Remove Member</a>.</li> </ul>
<code>xsuaa.patch.response.with.resource</code>	<p>Use this property if you want to retrieve a group whose membership was modified.</p> <div style="background-color: #f0f0f0; padding: 10px; margin: 10px 0;"> <p><b>Note</b></p> <p>This property is usable only when you have configured membership modifications via <a href="#">Add/Remove Member</a> UAA endpoints. That is, when the <code>scim.support.patch.operation</code> property is set to <b>false</b>.</p> </div> <p>Possible values:</p> <ul style="list-style-type: none"> <li><b>true</b> – the Identity Provisioning service will return the modified group via the <a href="#">GET /Groups</a> endpoint of UAA. To learn how, see <a href="#">Retrieve</a>.</li> <li><b>false</b> – no modified groups will be returned by the service.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://api.authentication.eu10.hana.ondemand.com
OAuth2TokenServiceURL=https://myaccount.authentication.eu10.hana.ondemand.com/oauth/token
User=MyXSUAUser
Password=*****
xsuaa.origin=myaccount-xsuaa.accounts.ondemand.com
xsuaa.origin.filter.enabled=true
scim.support.patch.operation=true
xsuaa.patch.response.with.resource=false
```

---

6. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP BTP XS Advanced UAA](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP BTP XS Advanced UAA system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Cloud Foundry UAA API: Users](#) ➡

[Cloud Foundry UAA API: Groups](#) ➡

Default read and write transformations:

→ Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints ([/Users](#) or [/Groups](#)) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external

application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$$.meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.name",
        "targetPath":
"$$.name",
        "optional": true
      },
      {
        "sourcePath":
"$$.emails",
        "targetPath":
"$$.emails",
        "preserveArrayWithSingleElement":
true
      },
      {
        "sourcePath":
"$$.emails[?(@.primary==
true)].value",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.groups",

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "constant": "xsuaa-dummy-
value",
        "targetPath": "$.id",
        "scope": "createEntity"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath": "$$.userName"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement":
true
      },
      {
        "condition": "$.emails[?
(@.primary == true)].value == []",
        "targetPath":
"$$.emails[0].primary",
        "constant": true
      },
      {
        "sourcePath":
"$$.phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.phoneNumbers",
        "optional": true
      },
      {
        "sourcePath":
"$$.externalId",
        "targetPath":
"$$.externalId",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },

```



```

    "targetPath":
    "$.groups",
    "preserveArrayWithSingleElement":
    true,
    "optional": true
  },
  {
    "sourcePath":
    "$.phoneNumbers",
    "targetPath":
    "$.phoneNumbers",
    "preserveArrayWithSingleElement":
    true,
    "optional": true
  },
  {
    "sourcePath":
    "$.active",
    "targetPath":
    "$.active",
    "optional": true
  },
  {
    "sourcePath":
    "$.meta",
    "targetPath":
    "$.meta",
    "optional": true
  },
  {
    "sourcePath":
    "$.externalId",
    "targetPath":
    "$.externalId",
    "optional": true
  },
  {
    "sourcePath":
    "$.origin",
    "targetPath":
    "$.origin",
    "optional": true
  },
  {
    "sourcePath":
    "$.zoneId",
    "targetPath":
    "$.zoneId",
    "optional": true
  },
  {
    "sourcePath":
    "$.verified",
    "targetPath":
    "$.verified",
    "optional": true
  },
  {

```

```

    {
      "sourcePath":
      "$.verified",
      "targetPath":
      "$.verified",
      "optional": true
    },
    {
      "constant":
      "%xsuaa.origin%",
      "targetPath": "$.origin"
    },
    {
      "constant":
      "urn:scim:schemas:core:1.0",
      "targetPath":
      "$.schemas[0]"
    },
    ],
    "scimEntityEndpoint": "Users"
  },
  "group": {
    "mappings": [
      {
        "sourceVariable":
        "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.displayName"
      },
      {
        "sourcePath":
        "$.description",
        "targetPath":
        "$.description",
        "optional": true
      },
      {
        "constant":
        "urn:scim:schemas:core:1.0",
        "targetPath":
        "$.schemas[0]"
      },
      {
        "sourcePath": "$.members",
        "targetPath": "$.members",
      }
    ]
  },
  "preserveArrayWithSingleElement":
  true,
  "optional": true
},
{
  "constant": "USER",
}
],
"preserveArrayWithSingleElement":
true,
"targetPath":
"$members[*].type",
"optional": true

```

```

        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:User",
        "targetPath":
"$$.schemas[0]"
    },
    "scimEntityEndpoint":
"Users"
    },
    "group": {
        "mappings": [
            {
                "sourcePath": "$$.id",
                "targetPath": "$$.id",
                "targetVariable":
"entityIdSourceSystem"
            },
            {
                "sourceVariable":
"entityBaseLocation",
                "targetPath":
"$$.meta.location",
                "targetVariable":
"entityLocationSourceSystem",
                "functions": [
                    {
                        "function":
"concatString",
                        "suffix": "$
{entityIdSourceSystem}"
                    }
                ]
            }
        ],
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
    },
    {
        "sourcePath":
"$$.description",
        "targetPath":
"$$.description",
        "optional": true
    },
    {
        "sourcePath":
"$$.members",
        "targetPath":
"$$.members",
        "preserveArrayWithSingleElement":
true,
        "optional": true
    },
    {
        "sourcePath":
"$$.zoneId",
        "targetPath":
"$$.zoneId"

```

```

    },
    {
        "sourcePath":
"$$.Operations",
        "targetPath":
"$$.Operations",
        "preserveArrayWithSingleElement":
true,
        "scope": "patchEntity"
    },
    {
        "sourcePath": "$$.schemas",
        "targetPath": "$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "scope": "patchEntity"
    },
    ],
    "scimEntityEndpoint": "Groups"
}

```

## Read Transformation

## Write Transformation

```

    },
    {
      "sourcePath":
"$ .meta",
      "targetPath":
"$ .meta",
      "optional": true
    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath":
"$ .schemas[0]"
    }
  ],
  "scimEntityEndpoint":
"Groups"
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

### SAP Cloud Identity Infrastructure

### Neo Environment

- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>• For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul> | <ul style="list-style-type: none"> <li>• For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul> |
|---|--|

### i Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation

is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

#### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[XS CLI: User Administration](#)

[Cloud Foundry UAA: Users](#) ➔

[Cloud Foundry UAA: Groups](#) ➔

### 1.6.3.11.1 Configure Single and Multiple Origins

Configure and provision users with single or multiple origins in SAP BTP XS Advanced UAA (Cloud Foundry) proxy system.

## Overview

An origin tells you which is the identity provider of a user in SAP BTP XS Advanced UAA (Cloud Foundry). It is defined in the trust configuration in the SAP BTP cockpit under ► [Security](#) ► [Trust Configuration](#) ► [Origin Key](#) 🔍.

The origin itself is not a concept of Identity Provisioning. The role of the service is to ensure that you can read and provision users with their identity providers. Once you find the identity provider in the Origin Key, you need to set it in the `xsuaa.origin` property. You can configure it in source, target and proxy SAP BTP XS Advanced UAA (Cloud Foundry) systems. Both single and multiple values are supported. The value is a string that usually specifies the name of the identity provider or its location.

For example: `xsuaa.origin=ldap` and `xsuaa.origin=ldap;myaccount-xsuaa.accounts.ondemand.com`, where the ";" (semicolon) is the only supported delimiter.

## Provisioning Users with Single Origin

You want to configure SAP BTP XS Advanced UAA (Cloud Foundry) as a proxy system for provisioning users with a single origin to and from an external identity management system.

1. On the [Properties](#) tab of the proxy system, set the `xsuaa.origin.filter.enabled` property to [true](#). This is a prerequisite for enabling the `xsuaa.origin` property in proxy systems.
2. Enter the value for the `xsuaa.origin` property, for example: [idp1](#). The value of this property acts like a filter.

Expect the following results:

- When executing a GET request, only users with the origin specified in the `xsuaa.origin` property are returned.
- When executing a POST request to create a user, you make one request to the proxy system. The payload contains the [idp1](#) origin.
- When executing a PUT request to update a user, you make one request to the proxy system. The payload contains the [idp1](#) origin.

## Provisioning Users with Multiple Origins

You want to configure SAP BTP XS Advanced UAA (Cloud Foundry) as a proxy system for provisioning users with multiple origins to and from an external identity management system.

1. On the [Properties](#) tab of the proxy system, set the `xsuaa.origin.filter.enabled` property to [true](#). This is a prerequisite for enabling the `xsuaa.origin` property in proxy systems.
2. Enter multiple values for the `xsuaa.origin` property, for example: [idp1;idp2](#). The value of this property acts like a filter.
3. On the [Transformations](#) tab, update the SAP BTP XS Advanced UAA (Cloud Foundry) proxy write transformation. Replace the constant and its value with the `sourcePath` expression pointing to a multivalue origin attribute:

### JSON Text Editor

Default mapping:

```
{
  "constant": "%xsuaa.origin%",
  "targetPath": "$.origin"
},
```

Expected mapping:

```
{
  "sourcePath": "$.origin",
  "targetPath": "$.origin"
},
```

### Graphical Editor

Default mapping:



Expected mapping:



4.
  - When executing a GET request, only users with the origins specified in the `xsuaa.origin` property are returned.
  - When executing a POST request to create a user with two origins (*idp1;idp2*), you should make two separate requests to the proxy system with two separate payloads - the first one containing the *idp1* origin and the second one containing *idp2* origin.  
As a result, two entries for one and the same user are created. Both entries contain the same user attributes (for example, username and email address) and differ only in their origins.
  - When executing a PUT request to update a user with two origins (*idp1;idp2*), you should make two separate requests to the proxy system with two separate payloads - the first one containing the *idp1* origin and the second one containing *idp2* origin.  
As a result, two entries for one and the same user are updated.

#### **i Note**

Multiple origins are not supported in provisioning scenarios between SAP BTP XS Advanced UAA (Cloud Foundry) source system and SAP BTP XS Advanced UAA (Cloud Foundry) target system.

## **1.6.3.12 SAP Build Work Zone, advanced edition**

Follow this procedure to set up SAP Build Work Zone, advanced edition as a proxy system.

### **Prerequisites**

- You have OAuth credentials for SAP Build Work Zone, advanced edition. To learn how, see [SAP Build Work Zone, advanced edition: Add an OAuth Client](#)


#### **i Note**

Administrators of bundle tenants on Neo environment should enable the *Manage OAuth Clients* permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### **Context**

After fulfilling the prerequisites, follow the procedure below to create a proxy SAP Build Work Zone, advanced edition system to load its users into an on-premise system and provision groups and new users back to SAP Build Work Zone, advanced edition.

#### **SCIM Filtering Support**

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to

users. If your system supports *native read filtering*, the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the *Read Transformation*, there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
*GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.emails[0].value",
  "targetPath": "$.emails[0].value",
  "optional": true
},
```

You also set the following filter in the *Properties* tab: *workzone.user.filter = addresses.country eq "US"*

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=emails[0].value eq "john.smith03@dummymail.com"**

The query request to the SAP Build Work Zone, advanced edition API will result into: **/Users?filter=addresses.country eq "US" and emails[0].value eq "john.smith03@dummymail.com"**

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

## Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li><li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li><li>3. Save your changes.</li><li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li><li>5. Save your changes.</li></ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li><li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li><li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li><li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li><li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li><li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li><li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li><li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li></ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Build Work Zone, advanced edition* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.



## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	<p>Enter the URL related to your SAP Build Work Zone, advanced edition system, in format: <b>https://&lt;account&gt;&lt;sap_wz_domain&gt;.workzone.ondemand.com</b></p> <p>For example: <i>https://mytenant.mydomain123.workzone.ondemand.com</i></p>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth client key, created for your SAP Build Work Zone, advanced edition tenant (see <b>Prerequisites</b> ).
Password	Enter the OAuth client secret, created for your SAP Build Work Zone, advanced edition tenant (see <b>Prerequisites</b> ).
OAuth2TokenServiceURL	<p>Enter the URL of the access token provider service for your SAP Build Work Zone, advanced edition instance, in format: <b>https://&lt;account&gt;&lt;sap_wz_domain&gt;.workzone.ondemand.com/api/v1/auth/token</b></p> <p>For example: <i>https://myaccount.mydomain123.workzone.ondemand.com/api/v1/auth/token</i></p>
Optional Properties	
(Optional) <code>workzone.content.type</code>	<p>This property makes the SAP Build Work Zone, advanced edition proxy system to send the specified value for the <i>Content-Type</i> HTTP header.</p> <p>Example: <b>application/json</b></p> <p>Default value (when not specified): <i>application/scim+json</i></p>

Property Name	Description & Value
(Optional) <code>workzone.support.patch.operation</code>	The default value of this property is <b>false</b> . But for SAP Build Work Zone, advanced edition proxy systems, this property appears during creation and its predefined value is <b>true</b> . That means, when the Identity Provisioning identifies a changed entity in the back-end system, it will execute the updates as PATCH requests instead of PUT. That means, only the changes will be written in SAP Build Work Zone, advanced edition, instead of provisioning the whole entity data.
(Optional) <code>workzone.user.unique.attribute</code>	<p>When the Identity Provisioning attempts to provision a user for the first time, it may detect that such a user already exists in SAP Build Work Zone, advanced edition. Thus, the service needs to retrieve the <i>entityId</i> of the existing user via filtering by user unique attribute(s). This property defines by which unique attribute(s) the existing user to be searched (resolved).</p> <p>Default value (when not specified): <i>userName</i></p> <p>To learn more, see: <a href="#">List of Properties</a></p>
(Optional) <code>workzone.group.unique.attribute</code>	<p>If the Identity Provisioning tries to create a group that already exists in SAP Build Work Zone, advanced edition, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific). To make the search filter by a specific attribute, specify this attribute as a value for this property.</p> <p>Default value (when not specified): <i>displayName</i></p> <p>To learn more, see: <a href="#">List of Properties</a></p>
(Optional) <code>workzone.include.if.match.wildcard.header</code>	<p>This property makes the SAP Build Work Zone, advanced edition target system to send the <i>If-Match</i> HTTP header with a value of "*" for every request to SAP Build Work Zone, advanced edition. This header could be used for entity versioning.</p> <p>Default value (when not specified): <i>false</i></p>
(Optional) <code>ips.failed.request.retry.attempts</code>	Predefined value: <i>2</i>
(Optional) <code>ips.failed.request.retry.attempts.interval</code>	Predefined value: <i>30</i>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the SAP Build Work Zone, advanced edition proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Build Work Zone, advanced edition. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Build Work Zone OData API](#) 

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourcePath":
"$ .id",
        "targetPath":
"$ .id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ],
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$ .emails[0].value",
        "targetPath":
"$ .emails[0].value",
        "optional": true
      },
      {
        "sourcePath":
"$ .emails[?(@.primary==
true)].value",
        "optional": true,
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$ .name",
        "optional": true,
        "targetPath":
"$ .name"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$ .id"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$ .id",
        "scope":
"deleteEntity"
      },
      {
        // When a user is supposed to be
        // deleted from SAP Build Work Zone,
        // advanced edition, it actually has
        // its status set to inactive instead
        // of being deleted.
        "constant": false,
        "targetPath":
"$ .active",
        "scope":
"deleteEntity"
      },
      {
        "sourcePath":
"$ .Operations",
        "targetPath":
"$ .Operations",
        "preserveArrayWithSingleElement":
true,
        "scope":
"patchEntity"
      },
      {
        "sourcePath":
"$ .schemas",
        "targetPath":
"$ .schemas",
        "preserveArrayWithSingleElement":
true,
        "scope":
"patchEntity"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$ .schemas[0]"
      }
    ]
  }
}

```

## Read Transformation

```

    {
      "sourcePath":
"$ .userType",
      "optional": true,
      "targetPath":
"$ .userType"
    },
    {
      "sourcePath":
"$ .displayName",
      "optional": true,
      "targetPath":
"$ .displayName"
    },
    {
      "sourcePath":
"$ .active",
      "optional": true,
      "targetPath":
"$ .active"
    },
    {
      "sourcePath":
"$ .title",
      "optional": true,
      "targetPath":
"$ .title"
    },
    {
      "sourcePath":
"$ .locale",
      "optional": true,
      "targetPath":
"$ .locale",
      "functions": [
        {
          "type":
"substring",
          "beginIndex": 0,
          "endIndex": 2
        }
      ]
    },
    {
      "sourcePath":
"$ .timezone",
      "optional": true,
      "targetPath":
"$ .timezone"
    },
    {
      "sourcePath":
"$ .addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$ .addresses"
    },

```

## Write Transformation

```

    {
      "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
      "targetPath":
"$ .schemas[1]"
    },
    {
      "sourcePath":
"$ .userType",
      "optional": true,
      "targetPath":
"$ .userType"
    },
    {
      "sourcePath":
"$ .schemas",
      "preserveArrayWithSingleElement":
true,
      "targetPath":
"$ .schemas",
      "optional": true
    },
    {
      "sourcePath":
"$ .userName",
      "targetPath":
"$ .userName"
    },
    {
      "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
[ 'userId' ]",
      "targetPath":
"$ .userName",
      "optional": true
    },
    {
      "sourcePath":
"$ .name",
      "targetPath":
"$ .name",
      "optional": true
    },
    {
      "sourcePath":
"$ .displayName",
      "targetPath":
"$ .displayName",
      "optional": true
    },
    {
      "sourcePath":
"$ .emails",
      "targetPath":
"$ .emails",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },

```

```

    {
      "sourcePath":
"$ .groups",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$ .groups"
    },
    {
      "sourcePath":
"$ .schemas",
      "preserveArrayWithSingleElement":
true,
      "targetPath":
"$ .schemas"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "optional": true
    }
  ]
}

```

```

    {
      "condition":
"$ .emails[0].length() > 0",
      "constant": true,
      "targetPath":
"$ .emails[0].primary"
    },
    {
      "targetPath":
"$ .locale",
      "type": "remove"
    },
    {
      "condition":
"($ .locale EMPTY false)
&& ($ .addresses[?(@.type ==
'work')].country EMPTY false)",
      "sourcePath":
"$ .locale",
      "targetPath":
"$ .locale",
      "functions": [
        {
          "function": "toLowerCaseString"
        },
        {
          "function": "concatString",
          "suffix":
"_"
        },
        {
          "function": "concatString",
          "suffix":
"$ .addresses[?(@.type ==
'work')].country"
        }
      ]
    },
    {
      "sourcePath":
"$ .timezone",
      "optional": true,
      "targetPath":
"$ .timezone"
    },
    {
      "sourcePath":
"$ .addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$ .addresses"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",

```

```

    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['displayName']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['displayName']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['userUuid']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['userUuid']",
      "optional": true
    }
  ],
  "group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [

```

```

      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
      "optional": true
    },
    {

```

```

    {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
    },
    {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
            {
                "type":
"concatString",
                "suffix":
"${entityIdSourceSystem}"
            }
        ],
        {
            "sourcePath":
"$$.displayName",
            "targetPath":
"$$.displayName"
        },
        {
            "sourcePath":
"$$.members",
            "preserveArrayWithSingleElement":
true,
            "optional": true,
            "targetPath":
"$$.members"
        },
        {
            "sourcePath":
"$$.schemas",
            "preserveArrayWithSingleElement":
true,
            "targetPath":
"$$.schemas"
        }
    ]
}

```

```

        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
        "optional": true
    }
]
},
"group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
        {
            "sourceVariable":
"entityIdTargetSystem",
            "targetPath":
"$$.id"
        },
        {
            "sourcePath":
"$$.Operations",
            "targetPath":
"$$.Operations",
            "preserveArrayWithSingleElement":
true,
            "scope":
"patchEntity"
        },
        {
            "sourcePath":
"$$.schemas",
            "targetPath":
"$$.schemas",
            "preserveArrayWithSingleElement":
true,
            "scope":
"patchEntity"
        },
        {
            "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
            "targetPath":
"$$.schemas[0]"
        },
        {
            "sourcePath":
"$$.schemas",
            "preserveArrayWithSingleElement":
true,
            "targetPath":
"$$.schemas",
            "optional": true
        },
        {

```



```

    "sourcePath":
    "$.displayName",
    "targetPath":
    "$.displayName"
  },
  {
    "sourcePath":
    "$.members",
    "preserveArrayWithSingleElement":
    true,
    "targetPath":
    "$.members",
    "optional": true
  }
]
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

#### Neo Environment

- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>• For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul> | <ul style="list-style-type: none"> <li>• For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul> |
|---|--|

#### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.13 SAP Build Work Zone, standard edition

Follow this procedure to set up SAP Build Work Zone, standard edition as a proxy system.

### Prerequisites

- You have created an instance and generated a service key for the standard service plan of SAP Build Work Zone, standard edition. For more information, see: [Initial Setup](#).  
The service key contains the API URL and the OAuth credentials (`clientid` and `clientsecret`) under the `uaa` property.

- **Note**

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context



The SAP Build Work Zone, standard edition simplifies access to SAP, custom-built, and third party applications and extensions (both on the cloud and on premise) by establishing a central launchpad.

You can use the Identity Provisioning UI to configure SAP Build Work Zone, standard edition as a proxy system and configure it in hybrid scenarios. For example, when SAP Build Work Zone, standard edition is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management and SAP Cloud Identity Access Governance, without making a direct connection between both systems. You can provision users, groups and users' group assignments to the external backend system, which can trigger CRUD (create, read, update, delete) operations on users, groups and users' group assignments back to the SAP Build Work Zone, standard edition.

This scenario supports provisioning of users and groups. In SAP Build Work Zone, standard edition, users can only be created and deleted. They cannot be updated as the update operation is skipped in the default proxy write transformation.

### Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit  [Neo](#)  [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Build Work Zone, standard edition* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Enter the SAP Build Work Zone, standard edition API URL from the service key of your SAP Build Work Zone, standard edition instance under <code>endpoints [portal-service]</code>. It follows the pattern:</p> <p><code>https://portal-service.cfapps.sap.hana.ondemand.com</code></p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client Id, from the service key of your SAP Build Work Zone, standard edition instance under <code>uaa.clientid</code> .
Password	(Credential) Enter the OAuth Client Secret, from the service key of your SAP Build Work Zone, standard edition instance under <code>uaa.clientsecret</code> .
OAuth2TokenServiceURL	<p>Enter the OAuth 2.0 Token Service URL from the service key of your SAP Build Work Zone, standard edition instance. It follows the pattern: <code>&lt;uaa.url&gt;/oauth/token</code>.</p> <p>For example: <code>https://ips-cflp-woaealle.authentication.sap.hana.ondemand.com/oauth/token</code></p>
(Optional) <code>cflp.providerId</code>	<p>Enter a valid providerID value.</p> <p>The provider ID is specified in the Channel Manager of the SAP Build Work Zone, standard edition when defining a new content provider. For more information about configuring the content provider to use the Identity Provisioning service, see: <a href="#">Manage Content Providers (Cloud)</a>, <a href="#">Manage Content Providers (On Premise)</a>.</p>

### Note

All users and groups are provisioned to the proxy SAP Build Work Zone, standard edition system with the providerID defined for this proxy system. If you want to use different providerIDs, you need to create separate proxy systems for every providerID.

Property Name	Value
(Optional) <code>cflp.user.filter</code>	<p>When specified, only those SAP Build Work Zone, standard edition users matching the filter expression will be read.</p> <p>By default, users are always filtered by the <code>providerId</code>. If another filtering attribute is defined, for example the email of the user, both filters are combined.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>emails.value eq 'john.smith@example.com'</code></li> </ul> <div> <p><b>Note</b></p> <p>Although, the email is supported as a filtering attribute, it is not returned when searching for the user.</p> </div> <ul style="list-style-type: none"> <li><code>urn:ietf:params:scim:schemas:extension:2.0:mapping.providerId eq 'ABC123'</code></li> </ul>
(Optional) <code>cflp.group.filter</code>	<p>When specified, only those SAP Build Work Zone, standard edition groups matching the filter expression will be read. By default, groups are always filtered by the <code>providerId</code>.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>externalId eq 12345678</code></li> <li><code>urn:ietf:params:scim:schemas:extension:2.0:mapping.providerId eq 'ABC123'</code></li> <li><code>meta.isIAG eq true</code> <p>This filtering attribute indicates whether the group will be used in a hybrid scenario with SAP Cloud Identity Access Governance.</p> </li> </ul>

Property Name	Value
<code>cflp.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>SAP Build Work Zone, standard edition supports the following unique attributes which are automatically filled in when the target system is added in the service UI:</p> <pre>emails[0].value, [ 'urn:ietf:params:scim:schemas:extension:2.0:mapping' ] [ 'providerId' ],externalId</pre> <ul style="list-style-type: none"> <li>• If the user has an <b>externalId</b>, the conflict is resolved by <b>externalId</b> and <b>providerId</b>.</li> <li>• If the user doesn't have an <b>externalId</b>, the conflict is resolved by <b>email</b> and <b>providerId</b>.</li> </ul> <p>For the conflict to be resolved, an existing user matching both unique attributes should be found. If an existing user doesn't match both unique attributes or matches only one of them, the user creation fails.</p> <div> <p>→ Recommendation</p> <p>We recommend that you do not modify the value of the <code>cflp.user.unique.attribute</code> property. Otherwise, user creation fails.</p> </div>

Property Name	Value
<code>cflp.group.unique.attribute</code>	<p>If Identity Provisioning tries to provision a group that already exists in the target system (a conflicting group), this property defines the unique attributes by which the existing group will be searched and resolved.</p> <p>SAP Build Work Zone, standard edition supports a pair of unique attributes which is automatically filled in when the target system is added in the service UI:</p> <pre><b>externalId,</b> <b>[ 'urn:ietf:params:scim:schemas:extension:2.0:mapping' ][ 'providerId' ]</b></pre> <p>For the conflict to be resolved, an existing group matching both unique attributes should be found. In this case, Identity Provisioning updates the group. This means, the conflicting group overwrites the existing one. If the group matches only one of the unique attributes, the conflict is not resolved, and the group creation fails.</p> <div> <p>→ Recommendation</p> <p>We recommend that you do not modify the value of the <code>cflp.group.unique.attribute</code> property. Otherwise, the group creation fails.</p> </div>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map user and group attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Build Work Zone, standard edition* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Build Work Zone, standard edition system. For more information, see: [Manage Transformations \[page 1494\]](#).

Default read and write transformations:

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem",
        "correlationAttribute": true
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$$.meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ],
        "sourcePath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.schemas"
      },
      {
        "sourcePath":
"$$.emails.value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.emails.value"
      },
      {
        "sourcePath":
"$$.externalId",
        "optional": true,
        "targetPath":
"$$.externalId"
      },
      {
        "sourcePath": "$$
[ 'urn:ietf:params:scim:schemas:ext

```

## Code Syntax

```

{
  "user": {
    "skipOperations": [
      "update"
    ],
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "sourcePath":
"$$.Operations",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.Operations",
        "scope":
"patchEntity"
      },
      {
        "sourcePath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.schemas",
        "scope":
"patchEntity"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$$.schemas[0]"
      },
      {
        "sourcePath":
"$$.emails",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.emails"
      },
      {
        "sourcePath": "$$
[ 'urn:ietf:params:scim:schemas:ext
[ 'providerId' ]",
        "optional": true,
        "targetPath": "$$
[ 'urn:ietf:params:scim:schemas:ext

```



```

extension:2.0:mapping']
['providerId']",
    {
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:2.0:mapping']
['providerId']"
    },
    {
        "sourcePath":
"$ .groups",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$ .groups"
    }
],
    {
        "scimEntityEndpoint":
"Users"
    },
    {
        "group": {
            "mappings": [
                {
                    "sourcePath":
"$ .id",
                    "targetPath":
"$ .id",
                    "targetVariable":
"entityIdSourceSystem",
                    "correlationAttribute": true
                },
                {
                    "sourceVariable":
"entityBaseLocation",
                    "targetPath":
"$ .meta.location",
                    "targetVariable":
"entityLocationSourceSystem"
                },
                {
                    "sourcePath":
"$ .schemas",
                    "preserveArrayWithSingleElement":
true,
                    "targetPath":
"$ .schemas"
                },
                {
                    "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:2.0:mapping']
['providerId']",
                    "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:2.0:mapping']
['providerId']"
                },
            ]
        }
    }
]

```

```

extension:2.0:mapping']
['providerId']"
    },
    {
        "condition": "($
['urn:ietf:params:scim:schemas:ext
ension:2.0:mapping']
['providerId'] EMPTY true)",
        "constant":
"%cflp.providerId%",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:2.0:mapping']
['providerId']"
    },
    {
        "sourcePath":
"$ .externalId",
        "optional": true,
        "targetPath":
"$ .externalId"
    },
    {
        "sourcePath":
"$ .groups[*].value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$ .groups[?(@.value)]"
    },
    {
        "scimEntityEndpoint":
"Users"
    },
    {
        "group": {
            "mappings": [
                {
                    "sourceVariable":
"entityIdTargetSystem",
                    "targetPath":
"$ .id"
                },
                {
                    "sourcePath":
"$ .Operations",
                    "preserveArrayWithSingleElement":
true,
                    "targetPath":
"$ .Operations",
                    "scope":
"patchEntity"
                },
                {
                    "sourcePath":
"$ .schemas",
                    "preserveArrayWithSingleElement":
true,

```

## Read Transformation

```

    "sourcePath":
    "$.meta.isIAG",
    "targetPath":
    "$.meta.isIAG"
    },
    {
        "condition":
        " '%cflp.providerId%' != 'null' ",
        "sourcePath":
        "$.externalId",
        "targetPath":
        "$.externalId"
    },
    {
        "condition":
        " '%cflp.providerId%' == 'null' ",
        "sourcePath":
        "$.externalId",
        "targetPath":
        "$.displayName"
    },
    {
        "sourcePath":
        "$.members",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetPath":
        "$.members"
    }
    ],
    "scimEntityEndpoint":
    "Groups"
    }
    }

```

## Write Transformation

```

    "targetPath":
    "$.schemas",
    "scope":
    "patchEntity"
    },
    {
        "constant":
        "urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath":
        "$.schemas[0]"
    },
    {
        "constant":
        "urn:ietf:params:scim:schemas:core:2.0:mapping",
        "targetPath":
        "$.schemas[1]"
    },
    {
        "sourcePath":
        "$.meta.isIAG",
        "optional": true,
        "targetPath":
        "$.meta.isIAG"
    },
    {
        "constant":
        "%cflp.providerId%",
        "targetPath": "$
        ['urn:ietf:params:scim:schemas:extension:2.0:mapping']
        ['providerId']"
    },
    {
        "sourcePath":
        "$.externalId",
        "optional": true,
        "targetPath":
        "$.externalId"
    },
    {
        "sourcePath":
        "$.members[*].value",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetPath":
        "$.members[?(@.value)]"
    }
    ],
    "scimEntityEndpoint":
    "Groups"
    }
    }

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)  
[Configure Integration with the Identity Provisioning Service](#)

### 1.6.3.14 SAP Business Network

Follow this procedure to set up SAP Business Network as a proxy system.


## Prerequisites

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

#### i Note

Currently, SAP Business Network connector is only available for selected customers who are approached by SAP. For more information, see [3305074](#) .

SAP Business Network, formerly known as Ariba Network, is a cloud-based offering that makes it possible for buyers and suppliers to collaborate on transactions, strengthen their relationships, and discover new business opportunities.

You can use Identity Provisioning to configure SAP Business Network as a proxy system in hybrid scenarios. For example, when SAP Business Network is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management, without making a direct connection between both systems. You can provision users and groups to the external backend system, which can trigger CRUD (create, read, update, delete) operations on users and group members back to the SAP Business Network.

#### i Note

Identity Provisioning cannot create and delete groups in SAP Business Network target system. It can only update existing groups by adding or removing group members. Therefore, groups must have been created in SAP Business Network.

## Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit ► [Neo](#) ► [Overview](#) ►, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li> <li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li> <li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li> <li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Business Network* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the SAP Business Network API URL.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client Id, created for your SAP Business Network system.
Password	(Credential) Enter the OAuth Client Secret, created for your SAP Business Network system.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.
bn.api.key	An API Key represents the unique key that identifies a particular application as a legitimate consumer of an API.
bn.realm.id	The realm name is part of the URL you use to access SAP Business Network.
(Optional) bn.user.filter	<p>When specified, only those SAP Business Network users matching the filter expression will be read. For example:</p> <ul style="list-style-type: none"><li>• <a href="#">userName eq "Julie Armstrong"</a></li><li>• <a href="#">userName sw "J"</a></li><li>• <a href="#">emails eq "julie.armstrong@example.com"</a></li></ul> <p>For more information, see <a href="#">List of Properties [page 94]</a></p>
(Optional) bn.group.filter	<p>When specified, only those SAP Business Network groups matching the filter expression will be read. For example:</p> <p><a href="#">displayName eq "Employees"</a></p> <p>For more information, see <a href="#">List of Properties [page 94]</a></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Business Network* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Business Network system. For more information, see:

[Manage Transformations \[page 1494\]](#).

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users* or */Groups*) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ]
[ 'userUuid' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ]
[ 'userUuid' ]",
        "optional": true
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ]
[ 'sendMail' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ]
[ 'sendMail' ]",
        "optional": true
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap.business.network:2.0:Us
er' ]['userStatus' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap.business.network:2.0:Us
er' ]['userStatus' ]",
        "optional": true
      },
      {
        "sourcePath":
"$$.externalId",
        "targetPath":
"$$.externalId",
        "correlationAttribute":
true,
        "optional": true
      },
      {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement":
true
      },
    ],
  },

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "constant": [
"urn:com.sap.ariba.framework.scim2.co
mmon.types.UserResourceV2",
"urn:ietf:params:scim:schemas:extensi
on:sap.business.network:2.0:User",
"urn:ietf:params:scim:schemas:core:2.
0:User",
"urn:ietf:params:scim:schemas:extensi
on:sap:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extens
ion:sap:2.0:User' ]['userUuid' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extens
ion:sap:2.0:User' ]['userUuid' ]"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extens
ion:sap.business.network:2.0:User' ]
[ 'userStatus' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extens
ion:sap.business.network:2.0:User' ]
[ 'userStatus' ]",
        "optional": true
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extens
ion:sap:2.0:User' ]['sendMail' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extens
ion:sap:2.0:User' ]['sendMail' ]",
        "optional": true
      },
      {
        "targetPath": "$.id",
        "sourceVariable":
"entityIdTargetSystem"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
    ],
  },

```



```

    {
      "sourcePath":
"$ .userName",
      "targetPath":
"$ .userName",
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .name.givenName",
      "targetPath":
"$ .name.givenName"
    },
    {
      "sourcePath":
"$ .name.middleName",
      "targetPath":
"$ .name.middleName",
      "optional": true
    },
    {
      "sourcePath":
"$ .name.familyName",
      "targetPath":
"$ .name.familyName"
    },
    {
      "sourcePath":
"$ .name.formatted",
      "targetPath":
"$ .name.formattedName",
      "optional": true
    },
    {
      "sourcePath":
"$ .name.honorificPrefix",
      "targetPath":
"$ .name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$ .name.honorificSuffix",
      "targetPath":
"$ .name.honorificSuffix",
      "optional": true
    },
    {
      "sourcePath":
"$ .displayName",
      "targetPath":
"$ .displayName",
      "optional": true
    },
    {
      "sourcePath": "$ .emails",
      "targetPath": "$ .emails",
    },
    "preserveArrayWithSingleElement":
true
  },

```

```

      "sourcePath":
"$ .name.givenName",
      "targetPath":
"$ .name.givenName"
    },
    {
      "sourcePath":
"$ .name.middleName",
      "targetPath":
"$ .name.middleName",
      "optional": true
    },
    {
      "sourcePath":
"$ .name.familyName",
      "targetPath":
"$ .name.familyName"
    },
    {
      "sourcePath":
"$ .name.honorificPrefix",
      "targetPath":
"$ .name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$ .name.honorificSuffix",
      "targetPath":
"$ .name.honorificSuffix",
      "optional": true
    },
    {
      "sourcePath": "$ .externalId",
      "targetPath": "$ .externalId",
      "optional": true
    },
    {
      "sourcePath":
"$ .displayName",
      "targetPath":
"$ .displayName",
      "optional": true
    },
    {
      "sourcePath":
"$ .emails[0].value",
      "targetPath":
"$ .emails[0].value"
    },
    {
      "sourcePath": "$ .active",
      "targetPath": "$ .active",
      "optional": true,
      "defaultValue": true
    },
    {
      "sourcePath": "$ .timezone",
      "targetPath": "$ .timezone",
      "optional": true
    },
    {
      "sourcePath":
"$ .preferredLanguage",

```

```

    {
      "sourcePath":
"$$.emails[0].value",
      "correlationAttribute":
true
    },
    {
      "sourcePath": "$$.active",
      "targetPath": "$$.active"
    },
    {
      "sourcePath":
"$$.preferredLanguage",
      "targetPath":
"$$.preferredLanguage",
      "optional": true
    },
    {
      "sourcePath":
"$$.addresses",
      "targetPath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath": "$$.locale",
      "targetPath": "$$.locale",
      "optional": true
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "targetPath":
"$$.phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.timezone",
      "targetPath":
"$$.timezone",
      "optional": true
    },
    {
      "sourcePath": "$$.meta",
      "targetPath": "$$.meta",
      "optional": true
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetPath":
"$$.meta.location",
      "targetVariable":
"entityLocationSourceSystem",
      "functions": [

```

```

      "targetPath":
"$$.preferredLanguage",
      "optional": true
    },
    {
      "sourcePath": "$$.locale",
      "targetPath": "$$.locale",
      "optional": true
    },
    {
      "condition": "$$.addresses[?
(@.type == 'work')].country EMPTY
false",
      "sourcePath": "$$.addresses[?
(@.type == 'work')]",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$$.addresses[0]"
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "targetPath":
"$$.phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath": "$$.Operations",
      "targetPath": "$$.Operations",
      "preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    }
  ],
  "group": {
    "scimEntityEndpoint": "Groups",
    "mappings": [
      {
        "targetPath": "$$.id",
        "sourceVariable":
"entityIdTargetSystem"
      },
      {
        "sourcePath": "$$.Operations",
        "targetPath": "$$.Operations",
        "preserveArrayWithSingleElement":
true,
        "scope": "patchEntity"
      },
      {
        "sourcePath": "$$.schemas",
        "targetPath": "$$.schemas",
        "preserveArrayWithSingleElement":
true,

```

```

        {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
        }
    ]
},
"group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
        {
            "sourcePath": "$.id",
            "targetVariable":
"entityIdSourceSystem",
            "targetPath": "$.id"
        },
        {
            "sourcePath": "$.schemas",
            "targetPath": "$.schemas",

"preserveArrayWithSingleElement":
true
        },
        {
            "sourcePath":
"$$.displayName",
            "targetPath":
"$$.displayName"
        },
        {
            "sourcePath": "$.members",
            "targetPath": "$.members",

"preserveArrayWithSingleElement":
true,
            "optional": true
        },
        {
            "sourcePath": "$.meta",
            "targetPath": "$.meta",
            "optional": true
        },
        {
            "sourceVariable":
"entityBaseLocation",
            "targetPath":
"$$.meta.location",
            "targetVariable":
"entityLocationSourceSystem",
            "functions": [
                {
                    "type":
"concatString",
                    "suffix": "$
{entityIdSourceSystem}"
                }
            ]
        }
    ]
}

```

```

        "scope": "patchEntity"
    },
    {
        "sourcePath":
"$$.displayName",
        "targetPath": "$.displayName"
    },
    {
        "sourcePath": "$.members",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.members",
        "optional": true
    }
]
}
}

```

## Read Transformation

## Write Transformation

```
    ]  
  }  
}
```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

### SAP Cloud Identity Infrastructure

### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
  - For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.
- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
  - For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[SAP Business Network](#)

### 1.6.3.15 SAP Central Business Configuration

Follow this procedure to set up SAP Central Business Configuration (in short, CBC) as a proxy system.

## Prerequisites

- You have created a technical user with administrator permissions that will be used to call the API of SAP Central Business Configuration for reading, creating and updating user and group member information.

### **i** Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

Create an SAP Central Business Configuration proxy connector to execute hybrid scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to the CBC system, whenever the external back-end requests such. This scenario supports provisioning **users** and **group members**.


### **⚠** Caution

You can't create or delete groups on CBC. That means:

- On the attempt to create a group on CBC, Identity Provisioning will only add new members or update existing ones. Also, if you read a group from the external back-end system, there must be a group with the exact same display name (case sensitive) in the CBC system. Otherwise, an error will be thrown and the group members will not be updated.

- On the attempt to delete a group on CBC, Identity Provisioning will only remove its members (group assignments). And this can happen only if the relevant group assignments have been provisioned/are present in the target system.

## SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
`GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'`
- If your system supports multivalued e-mails (that is `$.emails[0].value`, `$.emails[1].value`, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (`$.emails[0].value`).

## ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `cbc.user.filter = timezone eq "Africa"`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=userName eq "johnsmith03"`

The query request to the SAP Central Business Configuration API will result into: `/Users?filter=timezone eq "Africa" and userName eq "johnsmith03"`

## Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit ► [Neo](#) ► [Overview](#) ►, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li> <li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li> <li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li> <li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Central Business Configuration* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

## i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your CBC system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Specify the technical user for your CBC system.
Password	(Credential) Specify the password for your technical user.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.  For example: <b><a href="#">https://mycbcaccount.authentication.us2.hana.ondemand.com/oauth/token</a></b>
(Optional) <code>cbc.user.filter</code>	When specified, only those CBC users matching the filter expression will be read.  Example: <b><code>name.familyName eq "Smith" and addresses.country eq "US"</code></b>
(Optional) <code>cbc.group.filter</code>	When specified, only those CBC groups matching the filter expression will be read.  Example: <b><code>displayName eq "ProjectTeam1" or "Employees2020"</code></b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

### 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Central Business Configuration](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your CBC system. For more information, see [Manage Transformations \[page 1494\]](#).



Default read and write transformations:

→ Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
      {
        "sourcePath":
"$$.schema",
        "targetPath":
"$$.schema"
      },
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$$.meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
      },
      {
        "sourcePath":
"$$.members",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.members"
      }
    ],
    "user": {
      "scimEntityEndpoint":
"Users",
      "mappings": [
        {
          "sourcePath":
"$$.schemas",

```

## Code Syntax

```

{
  "group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
        "targetPath":
"$$.schemas[0]"
      },
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
      },
      {
        "sourcePath":
"$$.members[*].value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.members[?(@.value)]"
      },
      {
        "user": {
          "scimEntityEndpoint":
"Users",
          "mappings": [
            {
              "sourceVariable":
"entityIdTargetSystem",
              "targetPath":
"$$.id"
            },
            {
              "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
              "targetPath":
"$$.schemas[0]"
            },
            {
              "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
              "targetPath":
"$$.schemas[1]"
            },

```

## Read Transformation

## Write Transformation

```

"preserveArrayWithSingleElement":
true,
    "targetPath":
"$ .schemas"
    },
    {
        "sourcePath":
"$ .id",
        "targetPath":
"$ .id",
        "targetVariable":
"entityIdSourceSystem"
    },
    {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$ .meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
            {
                "type":
"concatString",
                "suffix":
"${entityIdSourceSystem}"
            }
        ]
    },
    {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "correlationAttribute": true
    },
    {
        "sourcePath":
"$ .name.givenName",
        "optional": true,
        "targetPath":
"$ .name.givenName"
    },
    {
        "sourcePath":
"$ .name.familyName",
        "optional": true,
        "targetPath":
"$ .name.familyName"
    },
    {
        "sourcePath":
"$ .active",
        "targetPath":
"$ .active"
    }
]
}

```

```

{
    "sourcePath":
"$ .userName",
    "targetPath":
"$ .userName"
},
{
    "sourcePath":
"$ .name.givenName",
    "targetPath":
"$ .name.givenName"
},
{
    "sourcePath":
"$ .name.familyName",
    "targetPath":
"$ .name.familyName"
},
{
    "sourcePath":
"$ .active",
    "targetPath":
"$ .active"
}
]
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"><li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li><li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li></ul>	<ul style="list-style-type: none"><li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li><li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li></ul>

#### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

#### Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[SAP Central Business Configuration – Collection](#) 

### 1.6.3.16 SAP Commerce Cloud

Follow this procedure to set up SAP Commerce Cloud as a proxy system.

#### Prerequisites

##### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.


- In SAP Commerce Cloud, you have added an OAuth client with authorization grant **Client Credentials**. To learn how, see: [Configuring OAuth Client](#).

#### Context

SAP Commerce Cloud is a highly flexible and modular platform for delivering modern customer experiences. It provides a standardized, automated, end-to-end solution that allows your projects to release code from repository to cloud.

You can use Identity Provisioning to configure SAP Commerce Cloud as a proxy system in hybrid scenarios. For example, when SAP Commerce Cloud is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management, without making a direct connection between both systems. You can provision users and groups to the external backend system, which can trigger CRUD (create, read, update, delete) operations on users and group back to the SAP Commerce Cloud.

##### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- 0 users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '[totalResults](#)' set to a value of 0.

- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the *Read Transformation*, there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
*GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the *Properties* tab: `scim.user.filter = emails.value eq "email@email.com"`

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP Commerce Cloud API will result into: **/Users?filter=emails.value eq "email@email.com" and userName eq "johnsmith03"**

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → *Neo* → *Overview*, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <i>Users &amp; Authorizations</i> ► <i>Administrators</i> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <i>Access Proxy System API</i> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <i>Security</i> ► <i>OAuth</i> ► <i>Clients</i> ► and choose <i>Register New Client</i>.</li> <li>2. From the <i>Subscription</i> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <i>Authorization Grant</i> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <i>Secret</i> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <i>Client ID</i>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <i>Subscriptions</i> ► <i>Java Applications</i> ► <i>ipsproxy</i> ►.</li> <li>7. From the left-side navigation, choose ► <i>Roles</i> ► <i>IPS_PROXY_USER</i> ►.</li> <li>8. Choose <i>Assign</i> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Commerce Cloud* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Enter the URL to your SAP Commerce Cloud system.</p> <p>The URL follows the pattern:  <a href="#">https://backoffice.&lt;tenant&gt;.model-t.cc.commerce.ondemand.com</a></p> <p>You can find the correct URL in the <a href="#">Environments</a> section of SAP Cloud Portal.</p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the client ID to retrieve the OAuth access token for SAP Commerce Cloud.
Password	(Credential) Enter the client secret to retrieve the OAuth access token for SAP Commerce Cloud.
OAuth2TokenServiceURL	<p>Enter the URL of the access token provider service for your SAP Commerce Cloud instance.</p> <p>This token URL is listed in the <a href="#">OAuth Clients</a> section of the <a href="#">App Integration</a> page.</p>
(Optional) <code>cc.user.filter</code>	<p>When specified, only those users matching the filter expression will be read. You can filter users by <b>userName</b>, <b>emails.value</b>, and <b>externalId</b>, according to the API syntax of SAP Commerce Cloud.</p> <p>For example:</p> <ul style="list-style-type: none"> <li><code>userName eq "johnbrown" and externalId eq "P000252"</code></li> <li><code>userName eq "johnbrown" and emails.value eq "johnbrown@email.com"</code></li> <li><code>userName eq "johnbrown" and emails.value eq "johnbrown@email.com" and externalId eq "P000252"</code></li> </ul> <div data-bbox="888 1592 979 1628" data-label="Section-Header"> <h3>Note</h3> </div> <div data-bbox="888 1646 1350 1715" data-label="Text"> <p>These combinations are valid for both 'or' and 'and' operators.</p> </div>
(Optional) <code>cc.group.filter</code>	<p>When specified, only those groups matching the filter expression will be read.</p> <p>For example:</p> <p><code>displayName eq "ProjectTeam1" or "Students2018"</code></p>



Property Name	Description & Value
(Optional) <code>cc.user.unique.attribute</code>	<p>This property defines by which unique attribute(s) an existing user in the target system will be searched and resolved in case Identity Provisioning tries to provision a conflicting user.</p> <p>SAP Commerce Cloud supports the following unique attributes which are automatically filled in during system creation: <code>emails[0].value</code>, <code>userName</code>, <code>externalId</code>.</p> <p>If the service finds an existing user by at least one of the unique attributes, it updates this user with the data of the conflicting one. If such a user is not found, the update of the conflicting user fails. If more than one users with these unique attributes are found, the update fails.</p>
(Optional) <code>cc.group.unique.attribute</code>	<p>If you try to provision a group that already exists in a target system, the group creation will fail. In this case, the existing group only needs to be updated.</p> <p>This property defines by which unique attribute(s) the existing group will be searched and resolved. The default value is <code>displayName</code>.</p> <p>If the service finds an existing group by this unique attribute, the group that you try to provision will overwrite the existing one. If such a group is not found, the group that you try to provision will not be created in the target system.</p>
(Optional) <code>cc.support.patch.operation</code>	<p>This property controls how modified entities (users and groups) in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>• If set to <code>true</code>, PATCH operations are used to update users and groups in the target system.</li> <li>• If set to <code>false</code>, PUT operations are used to update users and groups in the target system.</li> </ul> <p>Default value for proxy systems: <code>true</code></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Commerce Cloud* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Commerce Cloud. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Commerce Cloud SCIM Web Services API Documentation](#)

Default read and write transformations:

#### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$$.meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.schemas"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath": "$$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
        "optional": true,
        "targetPath": "$$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['userUuid']"
      },
      {
        "sourcePath":
"$$.externalId",
        "optional": true,

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName"
      },
      {
        "sourcePath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.schemas"
      },
      {
        "sourcePath": "$$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
        "optional": true,
        "targetPath": "$$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['userUuid']"
      },
      {
        "sourcePath":
"$$.externalId",
        "optional": true,
        "targetPath":
"$$.externalId"
      },
      {
        "sourcePath":
"$$.displayName",
        "optional": true,
        "targetPath":
"$$.displayName"
      },
      {
        "sourcePath":
"$$.nickName",
        "optional": true,
        "targetPath":
"$$.nickName"
      },
      {
        "sourcePath":
"$$.name.familyName",
        "optional": true,

```

```

    "targetPath":
    "$.externalId"
    },
    {
        "sourcePath":
        "$.displayName",
        "optional": true,
        "targetPath":
        "$.displayName"
    },
    {
        "sourcePath":
        "$.nickName",
        "optional": true,
        "targetPath":
        "$.nickName"
    },
    {
        "sourcePath":
        "$.name.familyName",
        "optional": true,
        "targetPath":
        "$.name.familyName"
    },
    {
        "sourcePath":
        "$.name.middleName",
        "optional": true,
        "targetPath":
        "$.name.middleName"
    },
    {
        "sourcePath":
        "$.name.givenName",
        "optional": true,
        "targetPath":
        "$.name.givenName"
    },
    {
        "sourcePath":
        "$.name.honorificSuffix",
        "optional": true,
        "targetPath":
        "$.name.honorificSuffix"
    },
    {
        "sourcePath":
        "$.addresses",
        "optional": true,
        "targetPath":
        "$.addresses"
    },
    {
        "sourcePath":
        "$.userType",
        "optional": true,
        "targetPath":
        "$.userType"
    },
    {

```

```

    "targetPath":
    "$.name.familyName"
    },
    {
        "sourcePath":
        "$.name.middleName",
        "optional": true,
        "targetPath":
        "$.name.middleName"
    },
    {
        "sourcePath":
        "$.name.givenName",
        "optional": true,
        "targetPath":
        "$.name.givenName"
    },
    {
        "sourcePath":
        "$.name.honorificSuffix",
        "optional": true,
        "targetPath":
        "$.name.honorificSuffix"
    },
    {
        "condition":
        "($.addresses[*].region EMPTY
        false) && ($.addresses[*].country
        EMPTY false)",
        "sourcePath":
        "$.addresses",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetPath":
        "$.addresses",
        "functions": [
            {
                "type":
                "convertCountryRegion",
                "outputFormat": "alpha2",
                "inputAttributes": [
                    "region",
                    "country"
                ],
                "outputAttribute": "region"
            }
        ],
        "sourcePath":
        "$.userType",
        "optional": true,
        "targetPath":
        "$.userType"
    },
    {

```

## Read Transformation

```

{
  "sourcePath":
  "$.active",
  "optional": true,
  "targetPath":
  "$.active"
},
{
  "sourcePath":
  "$.emails",
  "preserveArrayWithSingleElement":
  true,
  "optional": true,
  "targetPath":
  "$.emails"
},
{
  "sourcePath":
  "$.groups",
  "preserveArrayWithSingleElement":
  true,
  "optional": true,
  "targetPath":
  "$.groups"
},
{
  "scimEntityEndpoint":
  "Users"
},
{
  "group": {
    "mappings": [
      {
        "sourcePath":
        "$.id",
        "targetPath":
        "$.id",
        "targetVariable":
        "entityIdSourceSystem",
        "correlationAttribute": true
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetPath":
        "$.meta.location",
        "targetVariable":
        "entityLocationSourceSystem"
      },
      {
        "sourcePath":
        "$.schemas",
        "preserveArrayWithSingleElement":
        true,
        "targetPath":
        "$.schemas"
      }
    ]
  }
}

```

## Write Transformation

```

"sourcePath":
"$$.active",
"optional": true,
"targetPath":
"$$.active"
},
{
  "sourcePath":
  "$.emails",
  "preserveArrayWithSingleElement":
  true,
  "optional": true,
  "targetPath":
  "$.emails"
},
{
  "sourcePath":
  "$.groups[*].value",
  "preserveArrayWithSingleElement":
  true,
  "optional": true,
  "targetPath":
  "$.groups[?(@.value)]"
},
{
  "constant": "urn:ietf:params:scim:api:messages:2.0:PatchOp",
  "targetPath": "$.schemas[0]",
  "scope": "patchEntity"
},
{
  "sourcePath":
  "$.Operations",
  "preserveArrayWithSingleElement":
  true,
  "targetPath":
  "$.Operations",
  "scope":
  "patchEntity"
},
{
  "scimEntityEndpoint":
  "Users"
},
{
  "group": {
    "mappings": [
      {
        "sourceVariable":
        "entityIdTargetSystem",
        "targetPath":
        "$.id"
      },
      {
        "sourcePath":
        "$.displayName",
        "optional": true,
        "targetPath":
        "$.displayName"
      }
    ]
  }
}

```

## Read Transformation

```

    "sourcePath":
    "$.displayName",
    "targetPath":
    "$.displayName"
  },
  {
    "sourcePath":
    "$.members",
    "preserveArrayWithSingleElement":
    true,
    "optional": true,
    "targetPath":
    "$.members"
  },
  {
    "scimEntityEndpoint":
    "Groups"
  }
}

```

## Write Transformation

```

    },
    {
      "constant":
      "urn:ietf:params:scim:schemas:core:2.0:User",
      "targetPath":
      "$.schemas[0]"
    },
    {
      "sourcePath":
      "$.Operations",
      "preserveArrayWithSingleElement":
      true,
      "targetPath":
      "$.Operations",
      "scope":
      "patchEntity"
    },
    {
      "sourcePath":
      "$.schemas",
      "preserveArrayWithSingleElement":
      true,
      "targetPath":
      "$.schemas",
      "scope":
      "patchEntity"
    },
    {
      "sourcePath":
      "$.members[*].value",
      "preserveArrayWithSingleElement":
      true,
      "optional": true,
      "targetPath":
      "$.members[?(@.value)]"
    },
    {
      "scimEntityEndpoint":
      "Groups"
    }
  }
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

- Run an initial load job.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[SAP Cloud Identity Services Integration Architecture → cloudscimwebservices extension](#)

## 1.6.3.17 SAP Commissions

Follow this procedure to set up SAP Commissions as a proxy system.

### Prerequisites

- You have technical user credentials for an SAP Commissions system with read and write access permissions.


#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

Create a SCIM 2.0 proxy connector for SAP Commissions to execute hybrid scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to the SAP Commissions system, whenever the external back-end requests such. This scenario supports provisioning **users** and **user assignments to groups**.

#### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names ([<schema>:<attribute>](#)) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)



- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "targetVariable": "entityIdSourceSystem",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `scim.user.filter = name.familyName eq "Smith"`

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP Commissions API will result into: **/Users?filter=name.familyName eq "Smith" and userName eq "johnsmith03"**

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

#### SAP Cloud Identity Infrastructure

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

#### Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Commissions](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the SAP Commissions SCIM API portal.

Property Name	Value
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the user for your SAP Commissions system.
Password	Enter the password for your SAP Commissions user.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

Exemplary destination:

Type=[HTTP](#)

Authentication=[BasicAuthentication](#)

ProxyType=[Internet](#)

URL=<https://mycommissions.callidus.run/CallidusPortal>

User=[MyCommissionsUser](#)

Password=\*\*\*\*\*

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Commissions](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Commissions. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Commissions REST API](#) 

Default read and write transformations:

### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints ([/Users](#) or [/Groups](#)) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external

application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "targetVariable":
"entityIdSourceSystem",
        "correlationAttribute":
true
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$"
          }
        ]
      },
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "optional": true
      },
      {
        "sourcePath":
"$$.externalId",
        "targetPath":
"$$.externalId",
        "optional": true
      },
      {
        "sourcePath":
"$$.name.givenName",
        "targetPath":
"$$.name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$$.name.middleName",
        "targetPath":
"$$.name.middleName",
        "optional": true
      },
      {
        "sourcePath":
"$$.name.familyName",

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$$.userName",
        "targetPath": "$.id"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath": "$$.userName"
      },
      {
        "sourcePath":
"$$.externalId",
        "targetPath":
"$$.externalId",
        "optional": true,
        "defaultValue": ""
      },
      {
        "sourcePath":
"$$.name.givenName",
        "targetPath":
"$$.name.givenName",
        "optional": true,
        "defaultValue": ""
      },
      {
        "sourcePath":
"$$.name.familyName",
        "targetPath":
"$$.name.familyName",
        "optional": true,
        "defaultValue": ""
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath":
"$$.emails[0].value",
        "targetPath":
"$$.emails[0].value",
        "optional": true,
        "defaultValue": ""
      },
      {
        "condition": "$.emails[?
(@.primary == true)].value != []",
        "sourcePath": "$.emails[?
(@.primary == true)].value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,

```

```

      "targetPath":
"$ .name.familyName",
      "optional": true
    },
    {
      "sourcePath":
"$ .name.honorificPrefix",
      "targetPath":
"$ .name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$ .displayName",
      "targetPath":
"$ .displayName",
      "optional": true
    },
    {
      "sourcePath": "$ .active",
      "targetPath": "$ .active",
      "optional": true
    },
    {
      "sourcePath": "$ .emails",
      "targetPath": "$ .emails",
      "optional": true,

"preserveArrayWithSingleElement":
true
    },
    {
      "sourcePath":
"$ .emails[0].value",
      "targetPath":
"$ .emails[0].value",
      "optional": true
    },
    {
      "sourcePath": "$ .emails[?
(@.primary== true)].value",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .timezone",
      "optional": true,
      "targetPath": "$ .timezone"
    },
    {
      "sourcePath":
"$ .addresses",

"preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$ .addresses"
    },
    {

```

```

      "targetPath":
"$ .emails[0].value",
      "functions": [
        {
          "function":
"elementAt", "index": 0
        }
      ],
      {
        "constant": "",
        "targetPath":
"$ .groups[0].value"
      },
      {
        "condition": "$ .groups
EMPTY false",
        "sourcePath": "$ .groups",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$ .groups",
        "optional": true
      },
      {
        "sourcePath": "$ .locale",
        "targetPath": "$ .locale",
        "optional": true,
        "defaultValue": ""
      },
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName",
        "optional": true,
        "defaultValue": ""
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:User",
        "targetPath":
"$ .schemas[0]"
      }
    ],
    "group": {
      "scimEntityEndpoint":
"Groups",
      "skipOperations": [
        "create",
        "delete"
      ],
      "mappings": [
        {
          "sourcePath":
"$ .displayName",
          "targetPath": "$ .id"
        },
        {

```

```

        "sourcePath": "$.groups",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetPath": "$.groups"
    },
    {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement":
        true,
        "targetPath": "$.schemas"
    },
    {
        "sourcePath": "$.locale",
        "optional": true,
        "targetPath": "$.locale",
        "functions": [
            {
                "type": "substring",
                "beginIndex": 0,
                "endIndex": 2
            }
        ]
    }
]
},
"group": {
    "scimEntityEndpoint":
    "Groups",
    "mappings": [
        {
            "sourcePath":
            "$.displayName",
            "targetPath": "$.id",
            "targetVariable":
            "entityIdSourceSystem"
        },
        {
            "sourceVariable":
            "entityBaseLocation",
            "targetVariable":
            "entityLocationSourceSystem",
            "targetPath":
            "$.meta.location",
            "functions": [
                {
                    "type":
                    "concatString",
                    "suffix": "$
{entityIdSourceSystem}"
                }
            ]
        }
    ],
    {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.displayName"
    },
    {

```

```

        "sourcePath":
        "$.Operations",
        "targetPath":
        "$.Operations",
        "preserveArrayWithSingleElement":
        true,
        "scope": "patchEntity"
    },
    {
        "sourcePath": "$.schemas",
        "targetPath": "$.schemas",
        "preserveArrayWithSingleElement":
        true,
        "scope": "patchEntity"
    },
    {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.displayName"
    },
    {
        "optional": true,
        "preserveArrayWithSingleElement":
        true,
        "sourcePath": "$.members",
        "targetPath": "$.members"
    },
    {
        "constant":
        "urn:ietf:params:scim:schemas:core
:2.0:Group",
        "targetPath":
        "$.schemas[0]"
    }
]
}
}

```

```

        "sourcePath": "$.members",
        "targetPath": "$.members",

        "preserveArrayWithSingleElement":
        true,
            "optional": true
        },
        {
            "constant": "User",

        "preserveArrayWithSingleElement":
        true,
            "targetPath":
            "$.members[*].type",
            "optional": true
        },
        {
            "sourcePath": "$.schemas",
            "targetPath": "$.schemas",

        "preserveArrayWithSingleElement":
        true
        }
    ]
}
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

#### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.



## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[SAP Commissions: Integration with SAP IdP](#)

## 1.6.3.18 SAP Concur

Follow this procedure to set up SAP Concur as a proxy system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

User Provisioning Service (UPS) v4 API with [Pre-2017 Authorization](#)

- You have created a technical user with administrator permissions that will be used to call the UPS v4 API for creating or updating user account information. For more information, see [SAP Concur API: User Account Information](#).
- You have registered a partner application in your SAP Concur system. You need the administrator permissions to register the application. For more information, see [Registering a Partner Application](#).

#### User Provisioning Service (UPS) v4 API or SAP Concur Identity v4 API with *OAuth 2.0* authentication

- You have an SAP Concur admin user with *Web Services Administrator* role assigned.
- Your SAP Concur admin user has obtained a *Company Request Token* and a *Company UUID* from the SAP Concur Company Request Token self-service tool.  
For more information, see [Configure an SAP Concur Entity as an IdP Target](#) → *Section 2: SAP Concur Company Request Token*.

#### Note

Administrators of bundle tenants on Neo environment should enable the *Manage OAuth Clients* permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

Companies that use SAP Concur for managing and controlling travel expenses, invoices and other can use Identity Provisioning service to automate the identity and access management for the SAP Concur solution. You can use SAP Concur as a proxy connector to execute *hybrid* scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to SAP Concur, whenever the external back-end requests such. This scenario supports reading and writing of **users**.

SAP Concur offers three types of edition sites: **Standard**, **Professional** and **Standard-to-Professional Upgrade**. Its integration with Identity Provisioning service is supported with **Professional** edition only.

SAP Concur provides two APIs for its integration with Identity Provisioning: UPS v4 API and Identity v4 API (SCIM API). The value of the `concur.api.version` property controls which API you use.

- When the value is set to **1**, or the property is not defined (typical for systems created before versioning was introduced on December 8, 2021), UPS v4 API is used. The UPS v4 API currently supports two authentication methods: *Pre-2017 Authorization* and *OAuth 2.0*. For more information on how to update to version 2, see: [Update Connector Version \[page 1484\]](#)
- When the value is set to **2**, Identity v4 API is used. This is the value that Identity Provisioning automatically sets for newly created systems after versioning was introduced on December 8, 2021. Identity v4 API supports provisioning of users with `userUUID` attribute which is generated by Identity Authentication at user creation.

To create SAP Concur as a proxy system, proceed as follows:

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li><li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li><li>3. Save your changes.</li><li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li><li>5. Save your changes.</li></ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li><li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li><li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li><li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li><li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li><li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li><li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li><li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li></ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Concur* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the *Properties* tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Version 1 Mandatory Properties

Property Name	Description & Value
Type	Enter: <i>HTTP</i>
URL	Enter: <a href="https://www.concursolutions.com">https://www.concursolutions.com</a>
ProxyType	Enter: <i>Internet</i>
concur.api.version	Defines the version of SAP Concur API.  Set the value to <i>1</i> to use UPS v4 API.
Authentication	Enter: <i>BasicAuthentication</i>  When using UPS v4 API (Version 1), two types of <i>BasicAuthentication</i> are supported: <ul style="list-style-type: none"> <li>• <i>Pre-2017 Authorization</i> - Authentication based on Base-64 encoded Concur credentials (LoginID:Password) of the user. For more information, see <a href="#">Pre-2017 Authorization (Deprecated)</a>.</li> <li>• <i>OAuth 2.0</i> - For more information, see <a href="#">Getting Started</a>.</li> </ul>
User	Valid when <i>Pre-2017 Authorization</i> is used.  Enter the user ID of the SAP Concur technical user.
Password	Valid when <i>Pre-2017 Authorization</i> is used.  (Credential) Enter the password of the SAP Concur technical user.

Property Name	Description & Value
X-ConsumerKey	<p>Valid when <a href="#">Pre-2017 Authorization</a> is used.</p> <p>(Credential) Enter the key of the registered partner application (see the <b>Prerequisites</b> section).</p>
concur.datacenter	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Specify the SAP Concur data center your Identity Provisioning tenant belongs to. The following SAP Concur data centers are available:</p> <ul style="list-style-type: none"> <li>• us1</li> <li>• us2</li> <li>• eu1</li> <li>• eu2</li> <li>• emea</li> <li>• cn1</li> <li>• usg</li> <li>• int</li> </ul> <p>Based on the provided data center, Identity Provisioning configures the URL of the User Provisioning Service (UPS) v4 API or the SAP Concur Identity v4 API. For example, if you provide <b>us1</b>, the service will configure the URL in the following pattern: <code>us.api.concursolutions.com</code>.</p>

Property Name	Description & Value
<code>concur.authorization.code</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>(Credential) Enter the <a href="#">Company Request Token</a> and run a provisioning job within 24 hours from generating the token in the SAP Concur Company Request Token self-service tool. Otherwise, the token will expire, and you'll need a new one.</p> <p>After the first run of the job, Identity Provisioning fills in automatically a refresh token as the value of the <code>concur.refresh.token</code> property. If a provisioning job has not been run for six months, you'll again need to generate a new token.</p> <div> <p>→ Remember</p> <p>The company request token has a 24 hour validity. If this token expires, you must request a new token.</p> <p>The refresh token has a six month validity. Every time you run a provisioning job, the validity of the refresh token is extended with six months starting from the date of the last run. If you haven't run a provisioning job for six months, your refresh token will expire and you must request a new company request token.</p> </div>
<code>concur.company.id</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Enter the <a href="#">Company UUID</a> as described in the <i>Prerequisites</i> section.</p>
<code>concur.company.domain</code>	<p>Valid when the authentication type used is <a href="#">OAuth 2.0</a>.</p> <p>Enter your company domain.</p> <p>The username and the company domain are concatenated in the default transformation in the following format: <code>user@domain</code></p> <p>Your company domain is the part of your username behind the @ symbol. For example: <code>johnsmith@example.com</code></p>

#### Version 2 Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>

Property Name	Description & Value
<code>concur.datacenter</code>	<p>Specify the SAP Concur data center your Identity Provisioning tenant belongs to. The following SAP Concur data centers are available:</p> <ul style="list-style-type: none"> <li>• <code>us1</code></li> <li>• <code>us2</code></li> <li>• <code>eu1</code></li> <li>• <code>eu2</code></li> <li>• <code>emea</code></li> <li>• <code>cn1</code></li> <li>• <code>usg</code></li> <li>• <code>int</code></li> </ul> <p>Based on the provided data center, Identity Provisioning configures the URL of the User Provisioning Service (UPS) v4 API or the SAP Concur Identity v4 API. For example, if you provide <code>us1</code>, the service will configure the URL in the following pattern: <code>us.api.concursolutions.com</code>.</p>
<code>concur.api.version</code>	<p>Defines the version of SAP Concur API.</p> <p>Set the value to <code>2</code> to use Identity v4 API. This is the default value of the property.</p>
<code>concur.authorization.code</code>	<p>(Credential) Enter the <a href="#">Company Request Token</a> and run a provisioning job within 24 hours from generating the token in the SAP Concur Company Request Token self-service tool. Otherwise, the token will expire, and you'll need a new one.</p> <p>After the first run of the job, Identity Provisioning fills in automatically a refresh token as the value of the <code>concur.refresh.token</code> property. If a provisioning job has not been run for six months, you'll again need to generate a new token.</p> <div> <p>→ Remember</p> <p>The company request token has a 24 hour validity. If this token expires, you must request a new token.</p> <p>The refresh token has a six month validity. Every time you run a provisioning job, the validity of the refresh token is extended with six months starting from the date of the last run. If you haven't run a provisioning job for six months, your refresh token will expire and you must request a new company request token.</p> </div>

Property Name	Description & Value
<code>concur.company.id</code>	Enter the <a href="#">Company UUID</a> as described in the <i>Prerequisites</i> section.
<code>concur.company.domain</code>	<p>Enter your company domain.</p> <p>The username and the company domain are concatenated in the default transformation in the following format: <code>user@domain</code></p> <p>Your company domain is the part of your username behind the @ symbol. For example: <code>johnsmith@example.com</code></p>
(Optional) <code>concur.user.filter</code>	<p>When specified, only those users matching the filter expression will be read.</p> <p>For example:</p> <ul style="list-style-type: none"> <li><code>userName</code> eq "johnsmith@example.com" As the <code>userName</code> must be unique in SAP Concur, this filter returns only the user matching this <code>userName</code>.</li> <li><code>companyId</code> eq "aa067ada-71a9-4f57-8e98-9300b1c3171d" This filter returns all users in the company with this <code>companyId</code>.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Concur](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Concur. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: Concur \(Users\)](#) 

**UPS v4 API:** [User Provisioning Service v4 API](#) 

**Identity v4 API:** [Identity v4](#) 

### Caution

The UPS v4 API requires an initial password setup for all newly provisioned user accounts. The default transformation offers a statement with an empty string as a value for the password configuration. However, it is ignored in order to prevent from a default setup of a wrong initial password for your



systems. While the password statement is ignored, the provisioning will not be working. To enable the provisioning to SAP Concur, you need to perform the following operations:

1. Enable the password statement. To do this, either delete **"ignore": true**, or set it as **"ignore": false**.
2. Set a proper statement for the password attribute value (**"targetPath": "\$.Password"**).

(Optional) You can leave the default empty string, or you can use the **randomPassword** function to calculate a random value for the initial password of the newly created SAP Concur accounts. If you choose one of these two options and if you are not using single sign-on solution for SAP Concur, you have to also arrange a password reset support process in your company. This will securely offer an initial password to your corporate users for their newly created SAP Concur accounts. For more information, see [Transformation Expressions \[page 330\]](#) → **Transformation Functions**.

#### Default read and write transformation when using UPS v4 API (Version 1):

##### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy [Read Transformation](#) is used for [write](#) cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath":
        "$.EmployeeID",
        "targetPath":
        "$.id",
        "targetVariable":
        "entityIdSourceSystem",
        "functions": [
          {
            "type":
            "compositeId",
            "subId":
            "$.LoginID"
          }
        ],
        "sourcePath":
        "$.EmployeeID",
        "targetPath":
        "$.userName",
        "correlationAttribute": true
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [
          {
            "type":
            "concatString",
            "suffix":
            "${entityIdSourceSystem}"
          }
        ],
        "constant":
        "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
        "$.schemas[0]"
      },
      {
        "sourcePath":
        "$.PrimaryEmail",
        "targetPath":
        "$.emails[0].value",
        "correlationAttribute": true
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
        "$.userName",
        "targetPath":
        "$.EmpId"
      },
      {
        "sourcePath":
        "$.emails[0].value",
        "targetPath":
        "$.EmailAddress"
      },
      {
        "sourcePath":
        "$.emails[0].value",
        "targetPath":
        "$.LoginId"
      },
      {
        "sourcePath":
        "$.name.givenName",
        "targetPath":
        "$.FirstName"
      },
      {
        "sourcePath":
        "$.name.familyName",
        "targetPath":
        "$.LastName"
      },
      {
        "constant": "N",
        "targetPath":
        "$.Active"
      },
      {
        "condition":
        "$.active == true",
        "constant": "Y",
        "targetPath":
        "$.Active"
      },
      {
        "constant": "N",
        "targetPath":
        "$.ExpenseApprover"
      },
      {
        "constant": "N",
        "targetPath":
        "$.ExpenseUser"
      },
      {
        "constant": "N",

```

```

    },
    {
      "targetPath":
"$$.emails[0].primary",
      "constant": true
    },
    {
      "sourcePath":
"$$.FirstName",
      "optional": true,
      "targetPath":
"$$.name.givenName"
    },
    {
      "sourcePath":
"$$.LastName",
      "optional": true,
      "targetPath":
"$$.name.familyName"
    },
    {
      "sourcePath":
"$$.CellPhoneNumber",
      "optional": true,
      "targetPath":
"$$.phoneNumbers[0].value"
    },
    {
      "sourcePath":
"$$.LoginID",
      "correlationAttribute": true
    }
  ]
}

```

```

      "targetPath":
"$$.InvoiceApprover"
    },
    {
      "constant": "N",
      "targetPath":
"$$.InvoiceUser"
    },
    {
      "constant": "N",
      "targetPath":
"$$.IsTestEmp"
    },
    {
      "constant": "N",
      "targetPath":
"$$.TripUser"
    },
    {
      "ignore": true,
      "constant": "",
      "targetPath":
"$$.Password"
    },
    {
      "constant": "USD",
      "targetPath":
"$$.CrnKey"
    },
    {
      "constant": "US",
      "targetPath":
"$$.CtryCode"
    },
    {
      "sourcePath":
"$$.locale",
      "optional": true,
      "targetPath":
"$$.CtryCode",
      "functions": [
        {
          "type":
"substring",
          "beginIndex": 3
        }
      ],
      "constant":
"en_US",
      "targetPath":
"$$.LocaleName"
    },
    {
      "constant": "US",
      "targetPath":
"$$.Custom21"
    },
    {

```

## Read Transformation

### Write Transformation

```

"constant":
"DEFAULT",
"$ .LedgerName"
    },
    {
"constant":
"DEFAULT",
"$ .LedgerCode"
    }
]
}

```

## Default read and write transformation when using Identity v4 API (Version 2):

### Read Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath": "$.schemas"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$ .meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ],
        {
          "sourcePath": "$.active",
          "targetPath": "$.active"
        },
        {
          "sourcePath":
"$ .userName",
          "targetPath": "$ .userName"
        },
        {
          "sourcePath": "$.emails",
          "preserveArrayWithSingleElement":
true,
          "targetPath": "$.emails"
        },
        {
          "type": "remove",
          "targetPath":
"$ .emails[*].notifications"
        },
        {
          "sourcePath": "$.name",
          "targetPath": "$.name"
        }
      ]
    }
  }
}
```

### Write Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"@%concur.company.domain%"
          }
        ],
        {
          "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUid']",
          "optional": true,
          "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['userUid']"
        },
        {
          "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUid']",
          "optional": true,
          "targetPath":
"$ .externalId"
        },
        {
          "sourcePath": "$.emails",

```

```

    },
    {
      "sourcePath":
"$$.externalId",
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ]
[ 'userUid' ]",
      "optional": true
    },
    {
      "sourcePath":
"$$.externalId",
      "targetPath":
"$$.externalId",
      "optional": true
    },
    {
      "sourcePath":
"$$.displayName",
      "optional": true,
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath":
"$$.timezone",
      "optional": true,
      "targetPath": "$$.timezone"
    },
    {
      "sourcePath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$$.addresses"
    },
    {
      "sourcePath": "$$.title",
      "optional": true,
      "targetPath": "$$.title"
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$$.phoneNumbers"
    },
    {
      "sourcePath":
"$$.emergencyContacts",
      "targetPath":
"$$.emergencyContacts",

```

```

"preserveArrayWithSingleElement":
true,
      "targetPath": "$$.emails",
      "functions": [
        {
          "function":
"putIfAbsent",
          "key": "type",
          "defaultValue": "work"
        }
      ]
    },
    {
      "sourcePath": "$$.name",
      "targetPath": "$$.name"
    },
    {
      "sourcePath":
"$$.name.familyName",
      "targetPath":
"$$.name.familyName"
    },
    {
      "sourcePath":
"$$.name.givenName",
      "targetPath":
"$$.name.givenName"
    },
    {
      "sourcePath":
"$$.displayName",
      "optional": true,
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath":
"$$.timezone",
      "optional": true,
      "targetPath": "$$.timezone"
    },
    {
      "sourcePath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$$.addresses"
    },
    {
      "targetPath":
"$$.addresses[*].primary",
      "type": "remove"
    },
    {
      "sourcePath": "$$.title",
      "optional": true,
      "targetPath": "$$.title"
    },
    {

```

```

"preserveArrayWithSingleElement":
true,
    "optional": true
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'companyId' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'companyId' ]"
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]"
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'costCenter' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'costCenter' ]"
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'organization' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'organization' ]"
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'division' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'division' ]"
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext

```

```

        "sourcePath":
"$ .phoneNumbers[?(@.type !=
'mobile' || (@.type == 'mobile'
&& @.primary == false))]",
        "preserveArrayWithSingleElement":
true,
            "optional": true,
            "targetPath":
"$ .phoneNumbers[0]"
        },
        {
            "targetPath":
"$ .phoneNumbers[*].primary",
            "type": "remove"
        },
        {
            "sourcePath":
"$ .phoneNumbers[?(@.type ==
'mobile' && @.primary == true)]",
            "optional": true,
            "targetPath":
"$ .phoneNumbers[0]"
        },
        {
            "sourcePath":
"$ .emergencyContacts",
            "targetPath":
"$ .emergencyContacts",
            "preserveArrayWithSingleElement":
true,
                "optional": true
            },
            {
                "constant":
"%concur.company.id%",
                "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'companyId' ]"
            },
            {
                "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
                "optional": true,
                "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]"
            },
            {
                "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'costCenter' ]",
                "optional": true,
                "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'costCenter' ]"
            }

```

```

    ension:enterprise:2.0:User']
    ['department']",
        "optional": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['department']"
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['value']",
        "optional": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['value']"
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['displayName']",
        "optional": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['displayName']"
    }
    ]
}

```

```

    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['organization']",
        "optional": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['organization']"
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['division']",
        "optional": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['division']"
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['department']",
        "optional": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['department']"
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['value']",
        "optional": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['value']",
        "functions": [
            {
                "function":
                "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['displayName']",
        "optional": true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['displayName']"
    }
}

```



Read Transformation	Write Transformation
	<pre>     }   } ]</pre>

The Identity Provisioning does not provide group resource mapping in the SAP Concur write transformation. Although you cannot provision groups to SAP Concur, you can group the users into organizational units by enhancing your target transformation. For more information, see: [SAP Concur \(Target System\) \[page 817\]](#)

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### i Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

### 1.6.3.19 SAP CPQ

Follow this procedure to set up SAP CPQ as a proxy system.

## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- You have created a technical user with administrator permissions that will be used to call the API of SAP CPQ for reading, creating and updating user and group information.
- Make sure all users from the external consumer system that have been not provisioned via Identity Provisioning and you would like later to be written in SAP CPQ, have an organization unit name. This unit name must correspond to an existing company system ID in SAP CPQ. The [organization unit](#) and the [company system ID](#) must be exactly the same. Users with empty (missing) organization units will not be provisioned, as well as users whose organization units don't match any of the SAP CPQ company system IDs for the relevant tenant.
- In order for a created user to be active in SAP CPQ, it should be assigned to an SAP CPQ target group, whose ID ends with suffix `-USERTYPE`. To learn more, see [SAP CPQ: SCIM API](#) → section **Mappings between SCIM API and SAP CPQ** → **groups**.

## Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context


Create an SAP CPQ proxy connector to execute hybrid scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to the SAP CPQ system, whenever the external back-end requests such. This scenario supports provisioning **users** and **groups**.

## Caution

You can't create or delete groups on SAP CPQ. That means:

- On the attempt to create a group on SAP CPQ, Identity Provisioning will only add new members or update existing ones. Also, if you read a group from the external back-end system, there must be a group with the exact same display name (case sensitive) in the SAP CPQ system. Otherwise, an error will be thrown and the group members will not be updated.
- On the attempt to delete a group on SAP CPQ, Identity Provisioning will only remove its members (group assignments). And this can happen only if the relevant group assignments have been provisioned/are present in the target system.

## SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use '**eq**' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used '**eq**' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)

- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `cpq.user.filter = timezone eq "Africa"`

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP CPQ API will result into: **/Users?filter=timezone eq "Africa" and userName eq "johnsmith03"**

## Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit [► Neo ► Overview ►](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP CPQ](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>

Property Name	Value
URL	Specify the URL to the API of your SAP CPQ system. It is the same as your SAP CPQ tenant URL. It must contain the domain name.  For example: <b>https://sample1234.mycpqdomain.com</b>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Specify the technical user for your SAP CPQ system. It must also contain the domain name, in format: <code>&lt;user_name&gt;#&lt;domain_name&gt;</code>  For example: <b>JohnSmith#MYCPQDOMAIN</b>
Password	(Credential) Specify the password for your technical user.
(Optional) <code>cpq.user.filter</code>	When specified, only those SAP CPQ users matching the filter expression will be read.  Example: <b>name.familyName eq "Smith" and addresses.country eq "US"</b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP CPQ* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP CPQ system. For more information, see:

[Manage Transformations \[page 1494\]](#).

[SAP CPQ: SCIM API](#)

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users* or */Groups*) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourcePath":
"$$.schemas",
        "targetPath":
"$$.schemas"
      },
      {
        "sourcePath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem",
        "correlationAttribute": true,
        "targetPath":
"$$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.name.givenName",
        "targetPath":
"$$.name.givenName"
      },
      {
        "sourcePath":
"$$.name.familyName",
        "targetPath":
"$$.name.familyName"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "targetPath":
"$$.id",
        "sourceVariable":
"entityIdTargetSystem"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$$.schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath":
"$$.schemas[1]"
      },
      {
        "sourcePath":
"$$.externalId",
        "optional": true,
        "targetPath":
"$$.externalId"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName"
      },
      {
        "sourcePath":
"$$.name.givenName",
        "targetPath":
"$$.name.givenName"
      },
      {
        "sourcePath":
"$$.name.familyName",
        "targetPath":
"$$.name.familyName"
      },
      {
        "sourcePath":
"$$.emails",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.emails"
      }
    ]
  }
}

```



```

        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']"
    },
    {
        "sourcePath":
"$ .active",
        "targetPath":
"$ .active"
    },
    {
        "sourcePath":
"$ .emails",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$ .emails"
    },
    {
        "sourcePath":
"$ .addresses",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$ .addresses"
    },
    {
        "sourcePath":
"$ .phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$ .phoneNumbers"
    },
    {
        "sourcePath":
"$ .groups",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$ .groups"
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['userUuid']"
    }
],
"group": {

```

```

        "sourcePath":
"$ .active",
        "targetPath":
"$ .active"
    },
    {
        "sourcePath":
"$ .addresses",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$ .addresses"
    },
    {
        "sourcePath":
"$ .phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$ .phoneNumbers"
    },
    {
        "constant": [],
        "targetPath":
"$ .groups",
        "scope":
"createEntity"
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']"
    },
    {
        "constant": false,
        "targetPath": "$
['urn:sap:cpq:scim:schemas:extensi
on:custom:2.0:User'] ['IsSsoUser']"
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
        "optional": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['userUuid']"
    }
],
"group": {
    "scimEntityEndpoint":
"Groups",

```

```

"scimEntityEndpoint":
"Groups",
  "mappings": [
    {
      "sourcePath":
"$ .schemas",
      "targetPath":
"$ .schemas"
    },
    {
      "sourcePath":
"$ .id",
      "targetVariable":
"entityIdSourceSystem",
      "correlationAttribute": true,
      "targetPath":
"$ .id"
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetVariable":
"entityLocationSourceSystem",
      "targetPath":
"$ .meta.location",
      "functions": [
        {
          "type":
"concatString",
          "suffix":
"${entityIdSourceSystem}"
        }
      ]
    },
    {
      "sourcePath":
"$ .displayName",
      "targetPath":
"$ .displayName"
    },
    {
      "sourcePath":
"$ .members",
      "preserveArrayWithSingleElement":
true,
      "targetPath":
"$ .members"
    }
  ]
}

```

```

"mappings": [
  {
    "targetPath":
"$ .id",
    "sourceVariable":
"entityIdTargetSystem"
  },
  {
    "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
    "targetPath":
"$ .schemas[0]"
  },
  {
    "sourcePath":
"$ .displayName",
    "targetPath":
"$ .displayName"
  },
  {
    "sourcePath":
"$ .Operations",
    "targetPath":
"$ .Operations",
    "preserveArrayWithSingleElement":
true,
    "scope":
"patchEntity"
  },
  {
    "sourcePath":
"$ .schemas",
    "targetPath":
"$ .schemas",
    "preserveArrayWithSingleElement":
true,
    "scope":
"patchEntity"
  },
  {
    "sourcePath":
"$ .members[*].value",
    "preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath":
"$ .members[?(@.value)]"
  }
]
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This

will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Next Steps

After the proxy connector is configured, a consumer application can start sending CRUD requests to it. That means, initially reading all (or a filtered set of) entities, and creating them in the SCIM 2.0 back-end of the consumer application. The proxy connector can continue read newly created or updated entities and provision them to the proxy system's back-end.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

**REMEMBER!** When creating a new user in the consumer application, you have to do the following mandatory settings:

- Enter [organization unit](#) for the user, because this is a mandatory attribute for SAP CPQ. An organization with that exact name must already exist in SAP CPQ, otherwise the new user will not be provisioned.
- In order the new user to be active, add a group privilege for it of type [-USERTYPE](#). To learn more, see [SAP CPQ: SCIM API](#) → section **Mappings between SCIM API and SAP CPQ** → `groups`.

### Caution

Effective September 2020, Neo tenants from Shanghai (China) can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

### 1.6.3.20 SAP Data Custodian

Follow this procedure to set up SAP Data Custodian as a proxy system.

#### Prerequisites

##### ! Restriction

This system is available for all standalone tenants and bundle tenants running on SAP Cloud Identity Services infrastructure. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.


- You have created an SAP Data Custodian tenant.
- You have created a user within your tenant with the required roles for your scenario.
- You have added your user to SAP Identity Service Management (SAP ISM).
- You have completed the Transparency and Control Service Onboarding Process or the Key Management Service Onboarding Process, depending on the scenario you want to implement. For more information, see [Transparency and Control Service Onboarding Process](#) and [Key Management Service Onboarding Process](#).

##### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

#### Context

##### i Note

Currently, SAP Data Custodian connector is only available for selected customers who are approached by SAP. For more information, see [3319946](#) .

SAP Data Custodian is a robust Software as a Service (SaaS) solution that protects sensitive data stored in public, private, hybrid, and multicloud environments. This solution integrates with partnered public hyperscalers, SAP applications, and SAP managed clouds.

You can use Identity Provisioning to configure SAP Data Custodian as a proxy system to execute hybrid scenarios. For example, when SAP Data Custodian is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management, without making a direct connection

between both systems. You can provision entities to the external backend system, which can trigger CRUD (create, read, update, delete) operations back to the SAP Data Custodian. This scenario supports provisioning **users** and **user assignments to groups**.

SCIM API 2.0 does not support managing of group assignments via the SCIM user resource. The "groups" attribute of the user is read-only. This means that the user group assignments should be managed via the SCIM group resources using the "members" attribute (as it is defined by the SCIM standard).

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#) standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
`GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'`
- If your system supports multivalued e-mails (that is `$.emails[0].value`, `$.emails[1].value`, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (`$.emails[0].value`).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "targetVariable": "entityIdSourceSystem",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `dc.user.filter = displayName eq "Smith"`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=userName eq "johnsmith03"`

The query request to the SAP Data Custodian API will result into: `/Users?filter=displayName eq "Smith" and userName eq "johnsmith03"`

## Procedure

- Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

- Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li> <li>Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>Save your changes.</li> <li>Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li> <li>Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li> <li>From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li> <li>In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li> <li>From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li> <li>From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li> <li>Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

- Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
- Add SAP Data Custodian as a proxy system. For more information, see [Add a System \[page 1477\]](#).

5. Choose the *Properties* tab to configure the connection settings for your system.

#### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the URL to the SAP Data Custodian SCIM API portal.
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth client key, created for your SAP Data Custodian tenant.
Password	Enter the OAuth client secret, created for your SAP Data Custodian tenant.
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Data Custodian instance, in format: <b>https://&lt;SAP_Data_Custodian_datacenter&gt;/api/v1/auth/token</b>
dc.user.filter	<p>When specified, only those SAP Data Custodian users matching the filter expression will be read. You can filter users by <i>userName</i>, <i>displayName</i> or <i>externalId</i>.</p> <p><b>Possible values:</b></p> <p>For example: <i>userName eq "Smith.J"</i></p>

Property Name	Value
<code>dc.group.filter</code>	<p>This property filters groups by display name or externalId.</p> <p>When specified, only those groups matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">SAP_Data_Custodian_Auditor</a></li> <li>• <a href="#">SAP_Data_Custodian_Service_Admin</a></li> <li>• <a href="#">SAP_Data_Custodian_Key_Admin</a></li> <li>• <a href="#">SAP_Data_Custodian_Key_User</a></li> </ul> <p>For example: <code>displayName eq "SAP_Data_Custodian_Auditor"</code></p>
<code>dc.support.patch.operation</code>	<p>This property controls how modified users in the source system are updated in the target system.</p> <ul style="list-style-type: none"> <li>• If set to <a href="#">true</a>, PATCH operations are used to update users in the target system. This means, for example, that if a user attribute is modified, only this change will be provisioned and applied in the target system.</li> <li>• If set to <a href="#">false</a>, PUT operations are used to update users in the target system. This means, for example, that if a user attribute is modified, all user attributes are replaced in the target system, instead of updating only the modified ones.</li> </ul> <p>Users can be updated in the target system in various cases, such as:</p> <ul style="list-style-type: none"> <li>• In the source system, some user attributes are modified, or new attributes are added.</li> <li>• In the source system, a condition or a filter is set for users not to be read anymore.</li> <li>• A user is deleted from the source system.</li> </ul> <p>In the last two cases, it's possible to keep the entity in the target system – it will not be deleted but only disabled. To do this, use the <code>deleteEntity</code> scope in the transformation of your target or proxy system. See: <a href="#">Transformation Expressions [page 330]</a> → <code>deleteEntity</code>.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">true</a></li> <li>• <a href="#">false</a></li> </ul> <p>Default value: <a href="#">false</a></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.



## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the SAP Data Custodian proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

### **i** Note

When Identity Authentication is configured as a source system, the default transformation logic:

- Provisions only groups and user group assignments if they are part of the following predefined group list:
  - [SAP\\_Data\\_Custodian\\_Auditor](#)
  - [SAP\\_Data\\_Custodian\\_Service\\_Admin](#)
  - [SAP\\_Data\\_Custodian\\_Key\\_Admin](#)
  - [SAP\\_Data\\_Custodian\\_Key\\_User](#)
- Skips some of the attributes from the identity records.
- Sets primary email for `userName` of the user.

This way, the transformation logic ensures that the identity data, sent to the Identity Provisioning SCIM REST API, is consistent.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Data Custodian. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Data Custodian SCIM 2.0 API](#) 

Default read and write transformations:

### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (`/Users` or `/Groups`) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy [Read Transformation](#) is used for [write](#) cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourcePath":
"$$.schemas",
        "targetPath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true
      },
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName",
        "optional": true
      },
      {
        "sourcePath":
"$$.name",
        "targetPath":
"$$.name",
        "optional": true
      },
      {
        "sourcePath":
"$$.externalId",
        "optional": true,
        "targetPath":
"$$.externalId"
      },
      {
        "sourcePath":
"$$.active",
        "targetPath":
"$$.active"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "constant":
["urn:ietf:params:scim:schemas:core:2.0:User", "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User", "urn:ietf:params:scim:schemas:extension:sap:2.0:User", "urn:sap:cloud:scim:schemas:extension:custom:2.0:User"],
        "targetPath":
"$$.schemas"
      },
      {
        "condition":
"$$.externalId EMPTY false",
        "sourcePath":
"$$.externalId",
        "targetPath":
"$$.externalId"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName"
      },
      {
        "sourcePath":
"$$.displayName",
        "optional": true,
        "targetPath":
"$$.displayName"
      },
      {
        "sourcePath":
"$$.name.givenName",
        "optional": true,
        "targetPath":
"$$.name.givenName"
      },
      {
        "sourcePath":
"$$.name.familyName",
        "optional": true,
        "targetPath":
"$$.name.familyName"
      }
    ]
  }
}

```

```

    "sourcePath":
    "$.emails",
    "targetPath":
    "$.emails",
    "preserveArrayWithSingleElement":
    true,
    "optional": true
  },
  {
    "sourcePath":
    "$.groups",
    "targetPath":
    "$.groups",
    "preserveArrayWithSingleElement":
    true,
    "optional": true
  },
  {
    "sourceVariable":
    "entityBaseLocation",
    "targetVariable":
    "entityLocationSourceSystem",
    "targetPath":
    "$.meta.location",
    "functions": [
      {
        "type":
        "concatString",
        "suffix": "$"
      }
    ]
  },
  {
    "group": {
      "scimEntityEndpoint":
      "Groups",
      "mappings": [
        {
          "sourcePath":
          "$.schemas",
          "targetPath":
          "$.schemas",
          "preserveArrayWithSingleElement":
          true
        },
        {
          "sourcePath":
          "$.id",
          "targetPath":
          "$.id",
          "targetVariable":
          "entityIdSourceSystem"
        },
        {
          "sourcePath":
          "$.displayName",

```

```

    "sourcePath":
    "$.name.formatted",
    "optional": true,
    "targetPath":
    "$.name.formatted"
  },
  {
    "sourcePath":
    "$.name.honorificPrefix",
    "optional": true,
    "targetPath":
    "$.name.honorificPrefix"
  },
  {
    "sourcePath":
    "$.name.honorificSuffix",
    "optional": true,
    "targetPath":
    "$.name.honorificSuffix"
  },
  {
    "sourcePath":
    "$.name.middleName",
    "optional": true,
    "targetPath":
    "$.name.middleName"
  },
  {
    "sourcePath":
    "$.active",
    "optional": true,
    "targetPath":
    "$.active"
  },
  {
    "sourcePath":
    "$.emails",
    "preserveArrayWithSingleElement":
    true,
    "targetPath":
    "$.emails"
  },
  {
    "constant": "urn:ietf:params:scim:api:messages:2.0:PatchOp",
    "targetPath": "$.schemas[0]",
    "scope": "patchEntity"
  },
  {
    "sourcePath":
    "$.Operations",
    "preserveArrayWithSingleElement":
    true,
    "targetPath":
    "$.Operations",
    "scope":
    "patchEntity"
  }

```

## Read Transformation

```

    "targetPath":
    "$.displayName"
    },
    {
        "sourcePath":
        "$.members",
        "targetPath":
        "$.members",
        "optional": true,
        "preserveArrayWithSingleElement":
        true
    },
    {
        "sourcePath":
        "$.externalId",
        "targetPath":
        "$.externalId",
        "optional": true
    },
    {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [
            {
                "type":
                "concatString",
                "suffix": "$
                {entityIdSourceSystem}"
            }
        ]
    }
]
}

```

## Write Transformation

```

    ]
    },
    "group": {
        "scimEntityEndpoint":
        "Groups",
        "mappings": [
            {
                "sourceVariable":
                "entityIdTargetSystem",
                "targetPath":
                "$.id"
            },
            {
                "constant":
                "urn:ietf:params:scim:schemas:core:2.0:Group",
                "targetPath":
                "$.schemas[0]"
            },
            {
                "sourcePath":
                "$.displayName",
                "targetPath":
                "$.displayName"
            },
            {
                "sourcePath":
                "$.members",
                "preserveArrayWithSingleElement":
                true,
                "targetPath":
                "$.members",
                "optional": true
            },
            {
                "sourcePath":
                "$.Operations",
                "preserveArrayWithSingleElement":
                true,
                "targetPath":
                "$.Operations",
                "scope":
                "patchEntity"
            },
            {
                "sourcePath":
                "$.schemas",
                "targetPath":
                "$.schemas",
                "preserveArrayWithSingleElement":
                true,
                "scope":
                "patchEntity"
            }
        ]
    }
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"><li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li><li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li></ul>	<ul style="list-style-type: none"><li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li><li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li></ul>

#### i Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Related Information

[SAP Data Custodian](#)

### 1.6.3.21 SAP Fieldglass

Follow this procedure to set up a proxy connector for SAP Fieldglass.

## Prerequisites

- You have created an API application key and a web service. To do that, follow the steps on page: [Create API Application Key or Web Service](#) and [Web Services Setup](#)  
You will need the values of [Virtual Person Name \(Username\)](#) and [License Key](#) for the configuration of your proxy system (**step 6** below).

#### i Note


Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

You can use SAP Fieldglass as a proxy connector to execute *hybrid* scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to SAP Fieldglass, whenever the external back-end requests such. This scenario supports:

- Reading of **users** and **groups**
- Writing of **users** and **assignments**

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use '**eq**' filters by one SCIM attribute, and it's only applicable to users. If your system supports *native read filtering*, the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the *Read Transformation*, there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used '**eq**' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example:  
*GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

### Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the *Properties* tab: *scim.user.filter = timezone eq "US"*

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP Fieldglass API will result into: `/Users?filter=timezone eq "US" and userName eq "johnsmith03"`

Procedure

- 1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

- 2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Fieldglass](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>



Property Name	Value
URL	Specify your SAP Fieldglass environment URL. For example: <a href="https://abc123.fgvms.com">https://abc123.fgvms.com</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter your <a href="#">Virtual Person Name (Username)</a> – see the <b>Prerequisites</b> section above.
Password	(Credential) Enter your <a href="#">License Key</a> – see the <b>Prerequisites</b> section above.
OAuth2TokenServiceURL	Enter your OAuth token URL in the following format:  <code>https://&lt;Environment_URL&gt;/api/oauth2/v2.0/token</code>  For example: <a href="https://abc123.fgvms.com/api/oauth2/v2.0/token">https://abc123.fgvms.com/api/oauth2/v2.0/token</a>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Fieldglass](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Fieldglass. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: SAP Fieldglass](#) 

Default read and write transformations:

### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints ([/Users](#) or [/Groups](#)) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external

application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ],
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath": "$.name",
        "optional": true,
        "targetPath": "$.name"
      },
      {
        "sourcePath":
"$$.displayName",
        "optional": true,
        "targetPath":
"$$.displayName"
      },
      {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.title",
        "optional": true,
        "targetPath": "$.title"
      },
      {
        "sourcePath": "$.locale",
        "optional": true,

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "scope": "createEntity",
        "sourcePath":
"$$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName",
        "optional": true
      },
      {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "optional": true
      },
      {
        "sourcePath": "$.title",
        "targetPath": "$.title",
        "optional": true
      },
      {
        "sourcePath": "$.locale",
        "targetPath": "$.locale",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",

```

```

"preserveArrayWithSingleElement":
true
      },
      {
        "sourcePath":
"$$.emails[0].value",
        "targetPath":
"$$.emails[0].value"
      },
      {
        "optional": true,
        "defaultValue": "work",
        "sourcePath":
"$$.emails[0].type",

```

```

        "targetPath": "$.locale",
        "functions": [
            {
                "type": "substring",
                "beginIndex": 0,
                "endIndex": 2
            }
        ]
    },
    {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",

"preserveArrayWithSingleElement":
true
        },
        {
            "sourcePath":
"$$.emails[0].value",
            "targetPath":
"$$.emails[0].value"
        },
        {
            "sourcePath": "$.emails[?
(@.primary== true)].value",
            "correlationAttribute":
true
        },
        {
            "sourcePath":
"$$.timezone",
            "optional": true,
            "targetPath": "$.timezone"
        },
        {
            "sourcePath":
"$$.addresses",

"preserveArrayWithSingleElement":
true,
            "optional": true,
            "targetPath":
"$$.addresses"
        },
        {
            "sourcePath": "$.groups",

"preserveArrayWithSingleElement":
true,
            "optional": true,
            "targetPath": "$.groups"
        },
        {
            "sourcePath": "$.schemas",

"preserveArrayWithSingleElement":
true,
            "targetPath": "$.schemas"
        },
        {
            "sourcePath": "$
['resourceExtensions']

```

```
"targetPath":
"$$.emails[0].type"
    },
    {
        "defaultValue": true,
        "optional": true,
        "sourcePath":
"$$.emails[0].primary",
        "targetPath":
"$$.emails[0].primary"
    },
    {
        "sourcePath":
"$$.timezone",
        "optional": true,
        "targetPath": "$$.timezone"
    },
    {
        "sourcePath":
"$$.addresses",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.addresses"
    },
    {
        "sourcePath": "$$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath": "$$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
    },
    {
        "sourcePath": "$$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
        "targetPath": "$$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
        "optional": true
    },
    {
        "sourcePath": "$$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
        "targetPath": "$$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
        "optional": true
    },
    {
        "sourcePath": "$$
['urn:ietf:params:scim:schemas:ext
```

```
[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
    "optional": true
},
{
    "sourcePath": "$
['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'costCenter' ]",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'costCenter' ]",
    "optional": true
},
{
    "sourcePath": "$
['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'organization' ]",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'organization' ]",
    "optional": true
},
{
    "sourcePath": "$
['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'division' ]",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'division' ]",
    "optional": true
},
{
    "sourcePath": "$
['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'department' ]",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'department' ]",
    "optional": true
},
{
    "sourcePath": "$
['resourceExtensions']
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
```

```

    extension:enterprise:2.0:User']
    ['division']],
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['division']],
        "optional": true
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['department']],
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['department']],
        "optional": true
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager'] ['value']],
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager'] ['value']],
        "optional": true
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager'] ['displayName']],
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager'] ['displayName']],
        "optional": true
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:sap:2.0:User']
    ['userUuid']],
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:sap:2.0:User']
    ['userUuid']],
        "optional": true
    }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "skipOperations": [
            "create",
            "delete"
        ],
        "mappings": [
            {

```

```

    ension:enterprise:2.0:User']
    ['manager']['value'],
    "targetPath": "$
    ['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
    ['manager']['value'],
    "optional": true
  },
  {
    "sourcePath": "$
    ['resourceExtensions']
    ['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
    ['manager']['displayName'],
    "targetPath": "$
    ['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
    ['manager']['displayName'],
    "optional": true
  },
  {
    "sourcePath": "$
    ['resourceExtensions']
    ['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
    ['organization'],
    "targetPath": "$
    ['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
    ['organization'],
    "optional": true
  }
],
"group": {
  "scimEntityEndpoint":
  "Groups",
  "mappings": [
    {
      "sourcePath": "$.id",
      "targetPath": "$.id",
      "targetVariable":
      "entityIdSourceSystem"
    },
    {
      "sourceVariable":
      "entityBaseLocation",
      "targetVariable":
      "entityLocationSourceSystem",
      "targetPath":
      "$.meta.location",
      "functions": [
        {
          "type":
          "concatString",
          "suffix": "$
          {entityIdSourceSystem}"
        }
      ]
    }
  ],
  {

```

```

    "sourceVariable":
    "entityIdTargetSystem",
    "targetPath": "$.id"
  },
  {
    "sourcePath":
    "$.Operations",
    "targetPath":
    "$.Operations",
    "preserveArrayWithSingleElement":
    true,
    "scope": "patchEntity"
  },
  {
    "sourcePath": "$.schemas",
    "targetPath": "$.schemas",
    "preserveArrayWithSingleElement":
    true,
    "scope": "patchEntity"
  },
  {
    "sourcePath":
    "$.displayName",
    "targetPath":
    "$.displayName"
  },
  {
    "optional": true,
    "preserveArrayWithSingleElement":
    true,
    "sourcePath": "$.members",
    "targetPath": "$.members"
  }
]
}
}

```

```

        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
    },
    {
        "sourcePath": "$$.members",

"preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$$.members"
    },
    {
        "constant": "User",

"preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.members[*].type",
        "optional": true
    },
    {
        "sourcePath": "$$.schemas",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$$.schemas"
    }
]
}
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

### **i** Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### **⚠** Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)



## 1.6.3.22 SAP Field Service Management

Follow this procedure to set up SAP Field Service Management as a proxy system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

You have OAuth credentials for SAP Field Service Management. For more information, see: [Generating Client ID & Secret](#)

#### i Note

Administrators of bundle tenants on Neo environment should enable the *Manage OAuth Clients* permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

SAP Field Service Management is a cloud-based solution that is used to resolve customers issues with end-to-end field service management. For example, it helps customers overcome resource limitations, such as having enough skilled technicians in all locations.

You can use the Identity Provisioning user interface (UI) to configure SAP Field Service Management as a proxy system in hybrid scenarios. For example, when SAP Field Service Management is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management, without making a direct connection between both systems. You can provision users and groups to the external backend system, which can trigger CRUD (create, read, update, delete) operations on users and group members back to the SAP Field Service Management.

This scenario supports provisioning **users** and **group members**. When integrated with Identity Provisioning, SAP Field Service Management supports the following group concept:

- A user is created with a default group assigned. A group is mapped to a company (one of the key organizational units in SAP Field Service Management). When using the Identity Provisioning service API, the group is returned as follows:

```
<GroupName>_<CompanyName>
```

- A user can have one group assignment for each company. As in SAP Field Service Management a user can be assigned to multiple companies, this requires a group assignment for each of the companies the user can access. A user cannot be assigned to different groups mapped to one and the same company.

### ⚠ Caution

You cannot create or delete groups in SAP Field Service Management. In this case, you can expect the following behavior:

- If a new group is created in the source system (for example, Identity Authentication) and you run the provisioning job to SAP Field Service Management, the job will fail and no group will be created in the target system.
- If a group exists both in the source system (for example, Identity Authentication) and the target system - SAP Field Service Management, running a provisioning job will only add new members or update existing ones. In this case, groups in the source and target systems must have the same display name (case sensitive). Otherwise, the job will fail, and no group members will be updated.
- If a group exists in the target SAP Field Service Management system and you try to delete it, Identity Provisioning will only remove its group members. In this case, the relevant group members must exist in the target system.

## Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit ► [Neo](#) ► [Overview](#) ►, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

#### SAP Cloud Identity Infrastructure

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

#### Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Field Service Management](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.


### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>

Property Name	Value
URL	Specify the URL to the API of your SAP Field Service Management system. It follows the pattern:  <code>https://&lt;cluster&gt;.coresystems.net</code>
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the OAuth Client Id, created for your SAP Field Service Management system.
Password	(Credential) Enter the OAuth Client Secret, created for your SAP Field Service Management system.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.  For example: <code>https://&lt;fsm_account&gt;.coresuite.com/api/oauth2/v1/token</code>
(Optional) <code>fsm.user.filter</code>	When specified, only those SAP Field Service Management users matching the filter expression will be read.  Example: <b><code>name.familyName eq "Smith" and addresses.country eq "US"</code></b>
(Optional) <code>fsm.group.filter</code>	When specified, only those SAP Field Service Management groups matching the filter expression will be read.  Example: <b><code>displayName eq "ProjectTeam1" or "Employees2020"</code></b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports *native read filtering*, the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `scim.user.filter = timezone eq "Africa"`

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP Field Service Management API will result into: **/Users?filter=timezone eq "Africa" and userName eq "johnsmith03"**

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Field Service Management](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Field Service Management system. For more information, see:

[Manage Transformations \[page 1494\]](#).

[SAP Field Service Management - SCIM API](#)

Default read and write transformations:

#### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem",
        "correlationAttribute": true
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$$.meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ],
        "sourcePath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.schemas",
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.name.givenName",
        "optional": true,
        "targetPath":
"$$.name.givenName",
        "sourcePath":
"$$.name.familyName",
        "optional": true,
        "targetPath":
"$$.name.familyName"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id",
        "constant": [
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:ietf:params:scim:schemas:extension:sap:2.0:User"
        ],
        "targetPath":
"$$.schemas",
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "sourcePath":
"$$.name.givenName",
        "targetPath":
"$$.name.givenName",
        "sourcePath":
"$$.name.familyName",
        "targetPath":
"$$.name.familyName",
        "sourcePath":
"$$.active",
        "targetPath":
"$$.active",
        "sourcePath":
"$$.emails",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.emails",
        "sourcePath":
"$$.schemas:ext

```

```

    },
    {
      "sourcePath":
"$$.active",
      "targetPath":
"$$.active"
    },
    {
      "sourcePath":
"$$.emails",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$$.emails"
    },
    {
      "sourcePath":
"$$.groups",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$$.groups"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']['userUuid']"
    }
  ],
  "scimEntityEndpoint":
"Users"
},
{
  "group": {
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem",
        "correlationAttribute": true
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$$.meta.location",
        "targetVariable":
"entityLocationSourceSystem"
      }
    ]
  }
}

```

```

extension:sap:2.0:User']
['userUuid']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']['userUuid']"
    }
  ],
  "scimEntityEndpoint":
"Users"
},
{
  "group": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "sourcePath":
"$$.Operations",
        "targetPath":
"$$.Operations",
        "preserveArrayWithSingleElement":
true,
        "scope":
"patchEntity"
      },
      {
        "sourcePath":
"$$.schemas",
        "targetPath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,
        "scope":
"patchEntity"
      },
      {
        "targetPath":
"$$.id",
        "type": "remove",
        "scope":
"patchEntity"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
        "targetPath":
"$$.schemas[0]"
      },
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
      }
    ]
  }
}

```



## Read Transformation

```

    "sourcePath":
    "$.schemas",
    "preserveArrayWithSingleElement":
    true,
    "targetPath":
    "$.schemas"
    },
    {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.displayName"
    },
    {
        "sourcePath":
        "$.members",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetPath":
        "$.members"
    }
    ],
    "scimEntityEndpoint":
    "Groups"
    }
    }

```

## Write Transformation

```

    "sourcePath":
    "$.members[*].value",
    "preserveArrayWithSingleElement":
    true,
    "optional": true,
    "targetPath":
    "$.members[?(@.value)]"
    },
    "scimEntityEndpoint":
    "Groups"
    }
    }

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

## Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.23 SAP Integrated Business Planning for Supply Chain

Follow this procedure to set up SAP Integrated Business Planning for Supply Chain (in short, SAP IBP) as a proxy system.

## Prerequisites

- You have user credentials for an external back-end system with read and write permissions.
- To establish the connection between Identity Provisioning and SAP Integrated Business Planning for Supply Chain, you need to set up the communication (user, system and arrangement) on SAP Integrated Business Planning for Supply Chain. You can do it now (as a prerequisite) or in the process of configuring SAP Integrated Business Planning for Supply Chain as a proxy system, as described in step 5.

- **i Note**

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).


## Context

SAP Integrated Business Planning for Supply Chain is a cloud-based solution that combines sales and operations planning (S&OP), forecasting and demand, response and supply, demand-driven replenishment, and inventory planning.

You can use Identity Provisioning to configure SAP IBP as a proxy system to execute [hybrid](#) scenarios. For example, when SAP IBP is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management, without making a direct connection between both systems. You can provision entities to the external backend system, which can trigger CRUD (create, read, update, delete) operations back to SAP IBP. This scenario supports:

- Reading of **business users** (employee) and **business roles** (considered as [groups](#))
- Writing of **users** and **role assignments**

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

## ❖ Example

```
{
  "sourcePath": "$.user.userName",
  "targetPath": "$.userName",
  "optional": true,
  "correlationAttribute": true
},
```

Since [SAP IBP](#) doesn't support user filtering, then:

If, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "JOHNSMITH003"**

The query request to the SAP S/4HANA On-Premise API will result into a search for a user whose username is 'JOHNSMITH003'.

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

#### SAP Cloud Identity Infrastructure

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

#### Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Integrated Business Planning for Supply Chain](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Set up the communication between Identity Provisioning and SAP Integrated Business Planning for Supply Chain and configure your authentication method (basic or certificate-based).

### i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Integrated Business Planning for Supply Chain proxy system, select the [Certificate](#) tab and choose ► [Generate](#) ► [Download](#) ►, as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP Integrated Business Planning for Supply Chain backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide [User Name](#) and [Password](#).

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide *System ID*, *System Name* and *Host Name*.

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose *Scenario ID* SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

### **i** Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

6. Choose the *Properties* tab to configure the connection settings for your system.

### **i** Note


If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to your SAP IBP system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter your authentication method: <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div><b>! Restriction</b> Do not use special symbol ',' (comma) as it is not supported.</div>

Property Name	Description & Value
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
ibp.skip.read.archived	<p>In the event of archived (disabled) entities in your SAP IBP system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
ips.date.variable.format	<a href="#">yyyy-MM-dd</a>
ibp.user.roles.override	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP IBP proxy system in a hybrid scenario.</p> <ul style="list-style-type: none"> <li><b>true</b> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li><b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p>

Property Name	Description & Value
<code>ibp.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Integrated Business Planning for Supply Chain target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>personExternalID</code></li> <li><code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p>
(Optional) <code>ibp.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP IBP system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ <b>4.5 Filter System Query Option</b></p>
(Optional) <code>ibp.roles.page.size</code>	<p>Indicate how many business roles (considered as <code>groups</code>) per page to be read from your SAP IBP system.</p> <p>The value must be an integer number.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.



Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://my1234567-api.scmibp.ondemand.com
User=MyIBPuser
Password=*****
ips.date.variable.format=yyyy-MM-dd
ibp.skip.read.archived=true
ibp.user.roles.override=false
ibp.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true
ibp.roles.page.size=30
```

---

## 7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP IBP](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP IBP. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Integrated Business Planning API: Business User](#)

[SAP Business Accelerator Hub: SAP IBP](#) 

Default read and write transformations:

### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints ([/Users](#) or [/Groups](#)) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external

application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$$.personID",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ],
        {
          "sourcePath":
"$$.personalInformation.firstName",
          "targetPath":
"$$.name.givenName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.lastName",
          "targetPath":
"$$.name.familyName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.middleName",
          "targetPath":
"$$.name.middleName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.personFullName",
          "targetPath":
"$$.name.formatted",
          "optional": true
        },

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
        "optional": true,
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
[ 'userUuid' ]",
        "optional": true,
        "targetPath":
"$$.user.globalUserID"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.personID"
      },
      {
        "targetPath":
"$$.businessPartnerRoleCode",
        "type": "valueMapping",
        "sourcePaths": [
          "$$.userType"
        ],
        "defaultValue": "BUP003",
        "valueMappings": [
          {
            "key": [
              "Employee"
            ],
            "mappedValue":
"BUP003"
          }
        ],
        "scope": "createEntity",
        "sourceVariable":
"currentDate",
        "targetPath":
"$$.validityPeriod.startDate"
      },
      {
        "scope": "createEntity",
        "constant": "9999-12-31",

```

```

      "sourcePath":
"$ .user.userName",
      "targetPath":
"$ .userName",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "constant": true,
      "targetPath": "$ .active"
    },
    {
      "condition":
"$ .user.lockedIndicator ==
'true'",
      "constant": false,
      "targetPath": "$ .active",
      "optional": true
    },
    {
      "condition":
"($ .user.validityPeriod.startDate
> '{currentDate}') ||
('{currentDate}' >
$.user.validityPeriod.endDate)",
      "constant": false,
      "optional": true,
      "targetPath": "$ .active"
    },
    {
      "sourcePath":
"$ .workplaceInformation.emailAddress",
      "targetPath":
"$ .emails[0].value",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .user.logonLanguageCode",
      "optional": true,
      "targetPath": "$ .locale"
    },
    {
      "sourcePath":
"$ .PersonExternalID",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .user.globalUserID",
      "optional": true,
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUuid' ]"
    },

```

```

      "targetPath":
"$ .validityPeriod.endDate"
    },
    {
      "scope": "createEntity",
      "sourceVariable":
"currentDate",
      "targetPath":
"$ .user.validityPeriod.startDate"
    },
    {
      "scope": "createEntity",
      "constant": "9999-12-31",
      "targetPath":
"$ .user.validityPeriod.endDate"
    },
    {
      "sourcePath":
"$ .name.givenName",
      "optional": true,
      "targetPath":
"$ .personalInformation.firstName"
    },
    {
      "sourcePath":
"$ .name.familyName",
      "targetPath":
"$ .personalInformation.lastName"
    },
    {
      "sourcePath":
"$ .name.middleName",
      "optional": true,
      "targetPath":
"$ .personalInformation.middleName"
    },
    {
      "sourcePath":
"$ .name.formatted",
      "optional": true,
      "targetPath":
"$ .personalInformation.personFullName"
    },
    {
      "sourcePath":
"$ .userName",
      "targetPath":
"$ .user.userName"
    },
    {
      "sourcePath": "$ .locale",
      "optional": true,
      "targetPath":
"$ .user.logonLanguageCode"
    },
    {
      "sourcePath":
"$ .emails[0].value",
      "optional": true,
      "targetPath":
"$ .workplaceInformation.emailAddress"
    },

```

```

        "sourcePath":
        "$.user.role[*].roleName",

        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetPath": "$.groups[?
        (@.value)]"
    },
    {
        "type": "valueMapping",
        "sourcePaths": [
            "$.user.timeZoneCode"
        ],
        "targetPath":
        "$.timezone",
        "defaultValue": "Europe/
        Berlin",
        "valueMappings": [
            {
                "key": [
                    "WDFT"
                ],
                "mappedValue":
                "Europe/Berlin"
            },
            {
                "key": [
                    "ISRAEL"
                ],
                "mappedValue": "Asia/
                Jerusalem"
            },
            {
                "key": [
                    "RUS03"
                ],
                "mappedValue":
                "Europe/Moscow"
            },
            {
                "key": [
                    "AUSNSW"
                ],
                "mappedValue":
                "Australia/Sydney"
            },
            {
                "key": [
                    "UTC+4"
                ],
                "mappedValue": "Asia/
                Dubai"
            },
            {
                "key": [
                    "BRAZIL"
                ],
                "mappedValue":
                "America/Sao_Paulo"
            }
        ]
    }

```

```

    },
    {
        "condition": "$.active ==
        false",
        "constant": "true",
        "targetPath":
        "$.user.lockedIndicator"
    },
    {
        "scimEntityEndpoint": "Users"
    },
    {
        "group": {
            "mappings": [
                {
                    "sourcePath":
                    "$.displayName",
                    "targetVariable":
                    "entityIdTargetSystem",
                    "scope": "createEntity"
                },
                {
                    "sourcePath":
                    "$.displayName",
                    "targetPath":
                    "$.displayName"
                },
                {
                    "sourcePath":
                    "$.members[*].value",
                    "preserveArrayWithSingleElement":
                    true,
                    "optional": true,
                    "targetPath": "$.members[?
                    (@.value)]"
                },
                {
                    "sourcePath":
                    "$.Operations",
                    "targetPath":
                    "$.Operations",
                    "preserveArrayWithSingleElement":
                    true,
                    "scope": "patchEntity"
                },
                {
                    "sourcePath": "$.schemas",
                    "targetPath": "$.schemas",
                    "preserveArrayWithSingleElement":
                    true,
                    "scope": "patchEntity"
                }
            ],
            "scimEntityEndpoint": "Groups"
        }
    }

```

```

        "key": [
            "BRZLEA"
        ],
        "mappedValue":
"America/Sao_Paulo"
    },
    {
        "key": [
            "MSTNO"
        ],
        "mappedValue":
"America/Phoenix"
    },
    {
        "key": [
            "EST"
        ],
        "mappedValue":
"America/New_York"
    },
    {
        "key": [
            "UTC"
        ],
        "mappedValue": "Etc/
UTC"
    },
    {
        "key": [
            "UTC+3"
        ],
        "mappedValue": "Asia/
Riyadh"
    },
    {
        "key": [
            "EST_"
        ],
        "mappedValue":
"America/Toronto"
    },
    {
        "key": [
            "UTC+8"
        ],
        "mappedValue": "Asia/
Shanghai"
    },
    {
        "key": [
            "JAPAN"
        ],
        "mappedValue": "Asia/
Tokyo"
    }
    ]
},
{
    "type": "valueMapping",
    "sourcePaths": [
        "$.businessPartnerRoleCode"
    ]
}

```

```

    ],
    "targetPath":
"$ .userType",
    "defaultValue":
"Employee",
    "valueMappings": [
        {
            "key": [
                "BUP003"
            ],
            "mappedValue":
"Employee"
        }
    ]
},
"group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
        {
            "sourcePath": "$ .ID",
            "targetPath": "$ .id",
            "targetVariable":
"entityIdSourceSystem"
        },
        {
            "sourceVariable":
"entityBaseLocation",
            "targetVariable":
"entityLocationSourceSystem",
            "targetPath":
"$ .meta.location",
            "functions": [
                {
                    "type":
"concatString",
                    "suffix": "$
{entityIdSourceSystem}"
                }
            ]
        },
        {
            "sourcePath": "$ .ID",
            "targetPath":
"$ .displayName"
        },
        {
            "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
            "targetPath":
"$ .schemas[0]"
        },
        {
            "sourcePath":
"$ .to_BusinessUserAssignment.resul
ts",
            "optional": true,

```

```

"preserveArrayWithSingleElement":
true,
  "targetPath": "$.members"
},
{
  "type": "remove",
  "targetPath":
"$ .members[*].__metadata"
},
{
  "type": "remove",
  "targetPath":
"$ .members[*].UserName"
},
{
  "type": "rename",
  "constant": "value",
  "targetPath":
"$ .members[*].PersonID"
},
{
  "constant": "User",
  "targetPath":
"$ .members[*].type"
}
]
}
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource in the [Read Transformation](#). To learn how, see [Guided Answers: Business Role Description](#).

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.



### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.24 SAP Jam Collaboration

Follow this procedure to set up SAP Jam Collaboration as a proxy system.

### Prerequisites

- You get OAuth credentials for SAP Jam Collaboration. If your SAP Jam tenant is of "SCIM provisioning" type, an OAuth client is automatically created for it, with the name **SCIM API Client**. How to find this client:
  1. Go to the SAP Jam Collaboration admin panel.
  2. Choose ► [Integrations](#) ► [OAuth Clients](#) ►.
  3. For *SCIM API Client*, choose [View](#).
  4. Save the [Key](#) and [Secret](#) values – you'll need them later while configuring your SAP Jam Collaboration provisioning system.

To learn more, see: [SAP Jam: Add an OAuth Client](#)

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context


After fulfilling the prerequisites, follow the procedure below to create a proxy SAP Jam Collaboration system to load its users into an on-premise system and provision groups and new users back to SAP Jam.

#### ! Restriction

Bear in mind the following limitations for the number of sent requests during a provisioning job:

- The **SAP Jam SCIM API** allows up to 13,000 requests per hour and up to 200 requests per minute.
- The Identity Provisioning service can handle the 200 requests per minute limit. If more requests are sent during the minute, the service will "wait" until it can execute them.

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '[totalResults](#)' set to a value of 0.

- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the *Read Transformation*, there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
*GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

### ❁ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.emails[0].value",
  "targetPath": "$.emails[0].value",
  "optional": true
},
```

You also set the following filter in the *Properties* tab: *scim.user.filter = department eq "Management"*

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=emails[0].value eq "john.smith03@dummymail.com"**

The query request to the SAP Jam API will result into: **/Users?filter=department eq "Management" and emails[0].value eq "john.smith03@dummymail.com"**

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → *Neo* → *Overview*, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Jam Collaboration](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter the URL related to your SAP Jam system, in format: <a href="#">https://&lt;SAP_Jam_datacenter&gt;.sapjam.com</a> For example: <a href="#">https://jam4.sapjam.com</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth client key, created for your SAP Jam tenant (see <b>Prerequisites</b> ).
Password	Enter the OAuth client secret, created for your SAP Jam tenant (see <b>Prerequisites</b> ).
OAuth2TokenServiceURL	Enter the URL of the access token provider service for your SAP Jam instance, in format: <a href="#">https://&lt;SAP_Jam_datacenter&gt;/api/v1/auth/token</a> For example: <a href="#">https://jam4.sapjam.com/api/v1/auth/token</a>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Jam Collaboration](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Jam Collaboration. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Business Accelerator Hub: SAP Jam Collaboration](#) 

Default read and write transformations:

### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "sourcePath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ],
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.emails[0].value",
        "targetPath":
"$$.emails[0].value",
        "optional": true
      },
      {
        "sourcePath":
"$$.emails[?(@.primary==
true)].value",
        "optional": true,
        "correlationAttribute": true
      }
    ],
    "group": {
      "scimEntityEndpoint":
"Groups",

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id",
        "scope":
"deleteEntity"
      },
      {
        "condition":
"$$.emails[0].length() > 0",
        "constant": true,
        "targetPath":
"$$.emails[0].primary"
      },
      {
        "constant": false,
        "targetPath":
"$$.active",
        "scope":
"deleteEntity"
      },
      {
        "targetPath":
"$$.locale",
        "type": "remove"
      },
      {
        "condition":
"($$.locale EMPTY false)
&& ($$.addresses[?(@.type ==
'work')].country EMPTY false)",
        "sourcePath":
"$$.locale",
        "targetPath":
"$$.locale",
        "functions": [
          {
            "function": "toLowerCaseString"
          },
          {
            "function": "concatString",

```

```

    "mappings": [
      {
        "targetPath": "$",
        "sourcePath": "$"
      },
      {
        "sourcePath":
          "$.id",
        "targetVariable":
          "entityIdSourceSystem"
      },
      {
        "sourceVariable":
          "entityBaseLocation",
        "targetVariable":
          "entityLocationSourceSystem",
        "targetPath":
          "$.meta.location",
        "functions": [
          {
            "type":
              "concatString",
            "suffix":
              "${entityIdSourceSystem}"
          }
        ]
      }
    ]
  }
}

```

```

    "suffix":
      "-"
    },
    {
      "function": "concatString",
      "suffix":
        "$.addresses[?(@.type ==
        'work')].country"
    }
  ]
},
"group": {
  "scimEntityEndpoint":
    "Groups",
  "mappings": [
    {
      "sourcePath": "$",
      "targetPath": "$"
    },
    {
      "sourceVariable":
        "entityIdTargetSystem",
      "targetPath":
        "$.id"
    }
  ]
}
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### i Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.



## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

### 1.6.3.25 SAP Market Communication for Utilities

Follow this procedure to set up SAP Market Communication for Utilities as a proxy system.

## Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- You have user credentials for an external back-end system with read and write permissions.
- To establish the connection between Identity Provisioning and SAP Market Communication for Utilities, you need to set up the communication user in SAP Market Communication for Utilities. You can do it now (as a prerequisite) or in the process of configuring SAP Market Communication for Utilities as a proxy system, as described in step 5.
- **i Note**  
Administrators of bundle tenants on Neo environment should enable the *Manage OAuth Clients* permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

The SAP Market Communication for Utilities application is based on SAP BTP ABAP environment. You can use Identity Provisioning to configure SAP Market Communication for Utilities as a proxy system to execute *hybrid* scenarios. That means, it can provision its entities to another (external) back-end system by request, and then continue executing CRUD operations back to SAP Market Communication for Utilities, whenever the external back-end requests such. This scenario supports:

- Reading of **business users** (Employee) and **business roles** (which are considered as *groups*)
- Writing of **users** and **assignments**

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#) standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names ([<schema>:<attribute>](#)) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

#### ❖ Example

```
{
  "sourcePath": "$.user.userName",
  "targetPath": "$.userName",
  "optional": true,
  "correlationAttribute": true
},
```

Since SAP Market Communication for Utilities doesn't support user filtering, then:

If, for example, the SCIM Proxy endpoint request is: **GET /Users?filter=username eq "JOHNSMITH003"**

The query request to the SAP Market Communication for Utilities API will result into a search for a user whose username is 'JOHNSMITH003'.

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

## i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.


2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <a href="#">SAP Cloud Identity Infrastructure</a>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <a href="#">SAP BTP, Neo Environment</a>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li><li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li><li>3. Save your changes.</li><li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li><li>5. Save your changes.</li></ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"><li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li><li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li><li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li><li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li><li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li><li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li><li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li><li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li></ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Market Communication for Utilities](#) as a proxy system. For more information, see: [Add a System \[page 1477\]](#).
5. Set up the communication between Identity Provisioning and SAP Market Communication for Utilities and configure your authentication method (basic or certificate-based).

## i Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Market Communication for Utilities proxy system, select the [Certificate](#) tab and choose [Generate](#) > [Download](#) , as described in [Generate and Manage Certificates for Outbound Connection \[page 1507\]](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP Market Communication for Utilities backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide [User Name](#) and [Password](#).

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide [System ID](#), [System Name](#) and [Host Name](#).

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose [Scenario ID](#) SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

For more information, see [Maintain a Communication Arrangement for Inbound Communication](#) .

#### **i** Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

6. Choose the [Properties](#) tab to configure the connection settings for your system.

#### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.


If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the API URL to your SAP Market Communication for Utilities system.

Property Name	Description & Value
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"> <li>• <a href="#">BasicAuthentication</a></li> <li>• <a href="#">ClientCertificateAuthentication</a></li> </ul>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div> <p><b>! Restriction</b></p> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <p><b>! Restriction</b></p> <p>Do not use special symbol ',' (comma) as it is not supported.</p> </div>
maco.skip.read.archived	<p>In the event of archived (disabled) entities in your SAP Market Communication for Utilities system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
ips.date.variable.format	<p><a href="#">yyyy-MM-dd</a></p> <p>(needed for the <a href="#">Read Transformation</a>)</p>

Property Name	Description & Value
<code>maco.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP Market Communication for Utilities proxy system in a hybrid scenario.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p>
<code>maco.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Market Communication for Utilities target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>• Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>• Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• <code>personExternalID</code></li> <li>• <code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p>

Property Name	Description & Value
(Optional) <code>maco.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP Market Communication for Utilities system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ <b>4.5 Filter System Query Option</b></p>
(Optional) <code>maco.roles.page.size</code>	<p>Indicate how many business roles (considered as <a href="#">groups</a>) per page to be read from your SAP Market Communication for Utilities system.</p> <p>The value must be an integer number.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://12345-aaaaa-3333.abap.hana.ondemand.com
User=MyMaCoUser
Password=*****
ips.date.variable.format=yyyy-MM-dd
maco.skip.read.archived=true
maco.user.roles.override=false
maco.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true
maco.roles.page.size=30
```

## 7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SAP Market Communication for Utilities* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Market Communication for Utilities system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.



## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$$.personID",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ],
        {
          "sourcePath":
"$$.personalInformation.firstName",
          "targetPath":
"$$.name.givenName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.lastName",
          "targetPath":
"$$.name.familyName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.middleName",
          "targetPath":
"$$.name.middleName",
          "optional": true
        },
        {
          "sourcePath":
"$$.personalInformation.personFullName",
          "targetPath":
"$$.name.formatted",
          "optional": true
        },

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
        "optional": true,
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
[ 'userUuid' ]",
        "optional": true,
        "targetPath":
"$$.user.globalUserID"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.personID"
      },
      {
        "targetPath":
"$$.businessPartnerRoleCode",
        "type": "valueMapping",
        "sourcePaths": [
          "$.userType"
        ],
        "defaultValue": "BUP003",
        "valueMappings": [
          {
            "key": [
              "Employee"
            ],
            "mappedValue":
"BUP003"
          }
        ],
        "scope": "createEntity",
        "sourceVariable":
"currentDate",
        "targetPath":
"$$.validityPeriod.startDate"
      },
      {
        "scope": "createEntity",
        "constant": "9999-12-31",

```

```

        "sourcePath":
"$ .user.userName",
        "targetPath":
"$ .userName",
        "optional": true,
        "correlationAttribute":
true
    },
    {
        "targetPath":
"$ .schemas[0]",
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User"
    },
    {
        "targetPath":
"$ .schemas[1]",
        "constant":
"urn:ietf:params:scim:schemas:extension:sap:2.0:User"
    },
    {
        "constant": true,
        "targetPath": "$ .active"
    },
    {
        "condition":
"$ .user.lockedIndicator ==
'true'",
        "constant": false,
        "targetPath": "$ .active",
        "optional": true
    },
    {
        "condition":
"($ .user.validityPeriod.startDate
> '{currentDate}') ||
('{currentDate}' >
$.user.validityPeriod.endDate)",
        "constant": false,
        "optional": true,
        "targetPath": "$ .active"
    },
    {
        "sourcePath":
"$ .workplaceInformation.emailAddress",
        "targetPath":
"$ .emails[0].value",
        "optional": true,
        "correlationAttribute":
true
    },
    {
        "sourcePath":
"$ .user.logonLanguageCode",
        "optional": true,
        "targetPath": "$ .locale"
    },

```

```

        "targetPath":
"$ .validityPeriod.endDate"
    },
    {
        "scope": "createEntity",
        "sourceVariable":
"currentDate",
        "targetPath":
"$ .user.validityPeriod.startDate"
    },
    {
        "scope": "createEntity",
        "constant": "9999-12-31",
        "targetPath":
"$ .user.validityPeriod.endDate"
    },
    {
        "sourcePath":
"$ .name.givenName",
        "optional": true,
        "targetPath":
"$ .personalInformation.firstName"
    },
    {
        "sourcePath":
"$ .name.familyName",
        "targetPath":
"$ .personalInformation.lastName"
    },
    {
        "sourcePath":
"$ .name.middleName",
        "optional": true,
        "targetPath":
"$ .personalInformation.middleName"
    },
    {
        "sourcePath":
"$ .name.formatted",
        "optional": true,
        "targetPath":
"$ .personalInformation.personFullName"
    },
    {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .user.userName"
    },
    {
        "sourcePath":
"$ .nickName",
        "optional": true,
        "targetPath":
"$ .user.nickName"
    },
    {
        "sourcePath": "$ .locale",
        "optional": true,
        "targetPath":
"$ .user.logonLanguageCode"
    },

```

```

      "sourcePath":
"$ .PersonExternalID",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .user.role[*].roleName",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath": "$ .groups[?
(@.value)]"
    },
    {
      "sourcePath":
"$ .user.globalUserID",
      "optional": true,
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User' ][ 'userUuid' ]"
    },
    {
      "type": "valueMapping",
      "sourcePaths": [
        "$ .user.timeZoneCode"
      ],
      "targetPath":
"$ .timezone",
      "defaultValue": "Europe/
Berlin",
      "valueMappings": [
        {
          "key": [
            "WDFI"
          ],
          "mappedValue":
"Europe/Berlin"
        },
        {
          "key": [
            "ISRAEL"
          ],
          "mappedValue": "Asia/
Jerusalem"
        },
        {
          "key": [
            "RUS03"
          ],
          "mappedValue":
"Europe/Moscow"
        },
        {
          "key": [
            "AUSNSW"
          ],
          "mappedValue":
"Australia/Sydney"
        }
      ]
    }
  ]
}

```

```

    {
      "sourcePath":
"$ .emails[0].value",
      "optional": true,
      "targetPath":
"$ .workplaceInformation.emailAddre
ss"
    },
    {
      "condition": "$ .active ==
false",
      "constant": "true",
      "targetPath":
"$ .user.lockedIndicator"
    },
    {
      "scimEntityEndpoint": "Users"
    },
    {
      "group": {
        "mappings": [
          {
            "sourcePath":
"$ .displayName",
            "targetVariable":
"entityIdTargetSystem",
            "scope": "createEntity"
          },
          {
            "sourcePath":
"$ .displayName",
            "targetPath":
"$ .displayName"
          },
          {
            "sourcePath":
"$ .members[*].value",
            "preserveArrayWithSingleElement":
true,
            "optional": true,
            "targetPath": "$ .members[?
(@.value)]"
          },
          {
            "sourcePath":
"$ .Operations",
            "targetPath":
"$ .Operations",
            "preserveArrayWithSingleElement":
true,
            "scope": "patchEntity"
          },
          {
            "sourcePath": "$ .schemas",
            "targetPath": "$ .schemas",
            "preserveArrayWithSingleElement":
true,
            "scope": "patchEntity"
          }
        ]
      },
      "scimEntityEndpoint": "Groups"
    }
  ]
}

```

## Read Transformation

## Write Transformation

```

    {
      "key": [
        "UTC+4"
      ],
      "mappedValue": "Asia/
Dubai"
    },
    {
      "key": [
        "BRAZIL"
      ],
      "mappedValue":
"America/Sao_Paulo"
    },
    {
      "key": [
        "BRZLEA"
      ],
      "mappedValue":
"America/Sao_Paulo"
    },
    {
      "key": [
        "MSTNO"
      ],
      "mappedValue":
"America/Phoenix"
    },
    {
      "key": [
        "EST"
      ],
      "mappedValue":
"America/New_York"
    },
    {
      "key": [
        "UTC"
      ],
      "mappedValue": "Etc/
UTC"
    },
    {
      "key": [
        "UTC+3"
      ],
      "mappedValue": "Asia/
Riyadh"
    },
    {
      "key": [
        "EST_"
      ],
      "mappedValue":
"America/Toronto"
    },
    {
      "key": [
        "UTC+8"
      ],
      "mappedValue": "Asia/
Shanghai"
    }
  ]
}

```

```

}
}

```

```

        },
        {
            "key": [
                "JAPAN"
            ],
            "mappedValue": "Asia/
Tokyo"
        }
    ],
    },
    {
        "type": "valueMapping",
        "sourcePaths": [

            "$.businessPartnerRoleCode"
        ],
        "targetPath":
            "$.userType",
        "defaultValue":
            "Employee",
        "valueMappings": [
            {
                "key": [
                    "BUP003"
                ],
                "mappedValue":
                    "Employee"
            }
        ]
    }
],
},
"group": {
    "scimEntityEndpoint":
        "Groups",
    "mappings": [
        {
            "sourcePath": "$.ID",
            "targetPath": "$.id",
            "targetVariable":
                "entityIdSourceSystem"
        },
        {
            "sourceVariable":
                "entityBaseLocation",
            "targetVariable":
                "entityLocationSourceSystem",
            "targetPath":
                "$.meta.location",
            "functions": [
                {
                    "type":
                        "concatString",
                    "suffix": "$
{entityIdSourceSystem}"
                }
            ]
        }
    ],
    },
    {
        "sourcePath": "$.ID",
        "targetPath":
            "$.displayName"
    }

```

```

    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath":
"$$.schemas[0]"
    },
    {
      "sourcePath":
"$$.to_BusinessUserAssignment.results",
      "optional": true,

      "preserveArrayWithSingleElement":
true,
      "targetPath": "$$.members"
    },
    {
      "type": "remove",
      "targetPath":
"$$.members[*].__metadata"
    },
    {
      "type": "remove",
      "targetPath":
"$$.members[*].UserName"
    },
    {
      "type": "rename",
      "constant": "value",
      "targetPath":
"$$.members[*].PersonID"
    },
    {
      "constant": "User",
      "targetPath":
"$$.members[*].type"
    }
  ]
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource in the [Read Transformation](#). To learn how, see [Guided Answers: Business Role Description](#).

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.26 SAP Marketing Cloud

Follow this procedure to set up SAP Marketing Cloud as a proxy system.

### Prerequisites

- You have user credentials for an external back-end system with read and write permissions.
- To establish the connection between Identity Provisioning and SAP Marketing Cloud, you need to set up the communication (user, system and arrangement) on SAP Marketing Cloud. You can do it now (as a prerequisite) or in the process of configuring SAP Marketing Cloud as a proxy system, as described in step 5.

#### Note


Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

You can use SAP Marketing Cloud as a proxy connector to execute *hybrid* scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to SAP Marketing Cloud, whenever the external back-end requests such. This scenario supports:

- Reading of **users** and **Business roles** (which are considered as *groups*)
- Writing of **users** and **assignments**

#### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use '**eq**' filters by one SCIM attribute, and it's only applicable to users. If your system supports *native read filtering*, the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '*tooMany*'.



Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

#### ❖ Example

```
{
  "sourcePath": "$.user.userName",
  "targetPath": "$.userName",
  "optional": true,
  "correlationAttribute": true
},
```

Since [SAP Marketing Cloud](#) doesn't support user filtering, then:

If, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "JOHNSMITH003"**

The query request to the SAP Marketing Cloud API will result into a search for a user whose username is 'JOHNSMITH003'.

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

#### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► *Users & Authorizations* ► *Administrators* ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the *Access Proxy System API* permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► *Security* ► *OAuth* ► *Clients* ► and choose *Register New Client*.
2. From the *Subscription* combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the *Authorization Grant* combo box, select **Client Credentials**.
4. In the *Secret* field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated *Client ID*. You will need it later, too.
6. From the left-side navigation, choose ► *Subscriptions* ► *Java Applications* ► *ipsproxy* ►.
7. From the left-side navigation, choose ► *Roles* ► *IPS\_PROXY\_USER* ►.
8. Choose *Assign* and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Marketing Cloud* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Set up the communication between Identity Provisioning and SAP Marketing Cloud and configure your authentication method (basic or certificate-based).

### Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP Marketing Cloud proxy system, select the *Certificate* tab and choose ► *Generate* ► *Download* ►, as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP Marketing Cloud backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide *User Name* and *Password*.

For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide [System ID](#), [System Name](#) and [Host Name](#).

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose [Scenario ID](#) SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

### i Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

6. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note


If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to your SAP Marketing Cloud system.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter your authentication method: <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>

Property Name	Description & Value
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div> <b>! Restriction</b>  Do not use special symbol ',' (comma) as it is not supported. </div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b>  Do not use special symbol ',' (comma) as it is not supported. </div>
marketing.cloud.skip.read.archived	<p>In the event of archived (disabled) entities in your SAP Marketing Cloud system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
ips.date.variable.format	<p><a href="#">yyyy-MM-dd</a></p> <p>(needed for the <a href="#">Read Transformation</a>)</p>
marketing.cloud.user.roles.override	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP Marketing Cloud proxy system in a hybrid scenario.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p>

Property Name	Description & Value
<code>marketing.cloud.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP Marketing Cloud target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>personExternalID</code></li> <li><code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p>
(Optional) <code>marketing.cloud.roles.filter</code>	<p>Enter OData filtering for reading roles in the SAP Marketing Cloud system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a> ➔ <b>4.5 Filter System Query Option</b></p>
(Optional) <code>marketing.cloud.roles.page.size</code>	<p>Indicate how many business roles (considered as <code>groups</code>) per page to be read from your SAP Marketing Cloud system.</p> <p>The value must be an integer number.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://my1234567-api.s4hana.ondemand.com
User=MyMarketingCloudUser
Password=*****
ips.date.variable.format=yyyy-MM-dd
marketing.skip.read.archived=true
marketing.cloud.user.roles.override=false
marketing.cloud.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true
marketing.cloud.roles.page.size=30
```

---

7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Marketing Cloud](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP Marketing Cloud. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP Marketing Cloud API: Business User](#)

Default read and write transformations:

→ Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy [Read Transformation](#) is used for [write](#) cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$$.personID",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$$.personalInformation.firstName",
        "targetPath":
"$$.name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$$.personalInformation.lastName",
        "targetPath":
"$$.name.familyName",
        "optional": true
      },
      {
        "sourcePath":
"$$.personalInformation.middleName",
        "targetPath":
"$$.name.middleName",
        "optional": true
      },
      {
        "sourcePath":
"$$.personalInformation.personFullN
ame",
        "targetPath":
"$$.name.formatted",
        "optional": true
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true,
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
        "optional": true,
        "targetPath":
"$$.user.globalUserID"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.personID"
      },
      {
        "targetPath":
"$$.businessPartnerRoleCode",
        "type": "valueMapping",
        "sourcePaths": [
          "$.userType"
        ],
        "defaultValue": "BUP003",
        "valueMappings": [
          {
            "key": [
              "Employee"
            ],
            "mappedValue":
"BUP003"
          },
          {
            "key": [
              "Contingent Worker"
            ],
            "mappedValue":
"BBP005"
          }
        ]
      },
      {
        "scope": "createEntity",

```

```

      "sourcePath":
"$ .user.userName",
      "targetPath":
"$ .userName",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "constant": true,
      "targetPath": "$ .active"
    },
    {
      "condition":
"$ .user.lockedIndicator ==
'true'",
      "constant": false,
      "targetPath": "$ .active",
      "optional": true
    },
    {
      "condition":
"($ .user.validityPeriod.startDate
> '{currentDate}') ||
('{currentDate}' >
$.user.validityPeriod.endDate)",
      "constant": false,
      "optional": true,
      "targetPath": "$ .active"
    },
    {
      "sourcePath":
"$ .workplaceInformation.emailAddress",
      "targetPath":
"$ .emails[0].value",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .user.logonLanguageCode",
      "optional": true,
      "targetPath": "$ .locale"
    },
    {
      "sourcePath":
"$ .PersonExternalID",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .user.globalUserID",
      "optional": true,
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUid' ]"
    },

```

```

      "sourceVariable":
"currentDate",
      "targetPath":
"$ .validityPeriod.startDate"
    },
    {
      "scope": "createEntity",
      "constant": "9999-12-31",
      "targetPath":
"$ .validityPeriod.endDate"
    },
    {
      "scope": "createEntity",
      "sourceVariable":
"currentDate",
      "targetPath":
"$ .user.validityPeriod.startDate"
    },
    {
      "scope": "createEntity",
      "constant": "9999-12-31",
      "targetPath":
"$ .user.validityPeriod.endDate"
    },
    {
      "sourcePath":
"$ .name.givenName",
      "optional": true,
      "targetPath":
"$ .personalInformation.firstName"
    },
    {
      "sourcePath":
"$ .name.familyName",
      "targetPath":
"$ .personalInformation.lastName"
    },
    {
      "sourcePath":
"$ .name.middleName",
      "optional": true,
      "targetPath":
"$ .personalInformation.middleName"
    },
    {
      "sourcePath":
"$ .name.formatted",
      "optional": true,
      "targetPath":
"$ .personalInformation.personFullName"
    },
    {
      "sourcePath":
"$ .userName",
      "targetPath":
"$ .user.userName"
    },
    {
      "sourcePath": "$ .locale",
      "optional": true,
      "targetPath":
"$ .user.logonLanguageCode"

```



```

      "type": "valueMapping",
      "sourcePaths": [
        "$.user.timeZoneCode"
      ],
      "targetPath":
        "$.timezone",
      "defaultValue": "Europe/
Berlin",
      "valueMappings": [
        {
          "key": [
            "WDFT"
          ],
          "mappedValue":
            "Europe/Berlin"
        },
        {
          "key": [
            "ISRAEL"
          ],
          "mappedValue": "Asia/
Jerusalem"
        },
        {
          "key": [
            "RUS03"
          ],
          "mappedValue":
            "Europe/Moscow"
        },
        {
          "key": [
            "AUSNSW"
          ],
          "mappedValue":
            "Australia/Sydney"
        },
        {
          "key": [
            "UTC+4"
          ],
          "mappedValue": "Asia/
Dubai"
        },
        {
          "key": [
            "BRAZIL"
          ],
          "mappedValue":
            "America/Sao_Paulo"
        },
        {
          "key": [
            "BRZLEA"
          ],
          "mappedValue":
            "America/Sao_Paulo"
        },
        {
          "key": [
            "MSTNO"
          ],

```

```

      },
      {
        "sourcePath":
          "$.emails[0].value",
        "optional": true,
        "targetPath":
          "$.workplaceInformation.emailAddre
ss"
      },
      {
        "condition": "$.active ==
false",
        "constant": "true",
        "targetPath":
          "$.user.lockedIndicator"
      },
      {
        "scimEntityEndpoint": "Users"
      },
      {
        "group": {
          "mappings": [
            {
              "sourcePath":
                "$.displayName",
              "targetVariable":
                "entityIdTargetSystem",
              "scope": "createEntity"
            },
            {
              "sourcePath":
                "$.displayName",
              "targetPath":
                "$.displayName"
            },
            {
              "sourcePath":
                "$.members[*].value",
              "preserveArrayWithSingleElement":
                true,
              "optional": true,
              "targetPath": "$.members[?
(@.value)]"
            },
            {
              "sourcePath":
                "$.Operations",
              "targetPath":
                "$.Operations",
              "preserveArrayWithSingleElement":
                true,
              "scope": "patchEntity"
            },
            {
              "sourcePath": "$.schemas",
              "targetPath": "$.schemas",
              "preserveArrayWithSingleElement":
                true,
              "scope": "patchEntity"
            }
          ]
        }
      },
    ],

```

## Read Transformation

## Write Transformation

```

      "mappedValue":
"America/Phoenix"
    },
    {
      "key": [
        "EST"
      ],
      "mappedValue":
"America/New_York"
    },
    {
      "key": [
        "UTC"
      ],
      "mappedValue": "Etc/
UTC"
    },
    {
      "key": [
        "UTC+3"
      ],
      "mappedValue": "Asia/
Riyadh"
    },
    {
      "key": [
        "EST_"
      ],
      "mappedValue":
"America/Toronto"
    },
    {
      "key": [
        "UTC+8"
      ],
      "mappedValue": "Asia/
Shanghai"
    },
    {
      "key": [
        "JAPAN"
      ],
      "mappedValue": "Asia/
Tokyo"
    }
  ]
},
{
  "type": "valueMapping",
  "sourcePaths": [
    "$.businessPartnerRoleCode"
  ],
  "targetPath":
"$ .userType",
  "defaultValue":
"Employee",
  "valueMappings": [
    {
      "key": [
        "BUP003"
      ],

```

```

    "scimEntityEndpoint": "Groups"
  }
}

```

```

        "mappedValue":
"Employee"
        },
        {
            "key": [
                "BBP005"
            ],
            "mappedValue":
"Contingent Worker"
        }
    ]
},
{
    "sourcePath":
"$.user.role[*].roleName",
    "preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath": "$.groups[?
(@.value)]"
}
]
},
"group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
        {
            "sourcePath": "$.ID",
            "targetPath": "$.id",
            "targetVariable":
"entityIdSourceSystem"
        },
        {
            "sourceVariable":
"entityBaseLocation",
            "targetVariable":
"entityLocationSourceSystem",
            "targetPath":
"$.meta.location",
            "functions": [
                {
                    "type":
"concatString",
                    "suffix": "$
{entityIdSourceSystem}"
                }
            ]
        },
        {
            "sourcePath": "$.ID",
            "targetPath":
"$.displayName"
        },
        {
            "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
            "targetPath":
"$.schemas[0]"
        }
    ],

```

```

    {
      "sourcePath":
"$$.to_BusinessUserAssignment.results",
      "optional": true,
      "preserveArrayWithSingleElement":
true,
      "targetPath": "$.members"
    },
    {
      "type": "remove",
      "targetPath":
"$$.members[*].__metadata"
    },
    {
      "type": "remove",
      "targetPath":
"$$.members[*].UserName"
    },
    {
      "type": "rename",
      "constant": "value",
      "targetPath":
"$$.members[*].PersonID"
    },
    {
      "constant": "User",
      "targetPath":
"$$.members[*].type"
    }
  ]
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource in the [Read Transformation](#). To learn how, see [Guided Answers: Business Role Description](#).

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.27 SAP Master Data Integration

Follow this procedure to set up SAP Master Data Integration (in short, MDI) as a proxy system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- You have created a tenant in a subaccount on SAP BTP, Cloud Foundry environment. You can create your own tenant (free of charge), or integrate with an existing one.
- You have created a service instance for MDI in the subaccount in order to connect a new system to your tenant and read user account information from it. To learn how, see: [Creating Service Instances](#)
- You have created a service key in this instance, which contains the necessary credentials to connect to the MDI service. Creating multiple service keys in the same service instance is not supported. To learn how, see: [Creating service Instances](#)

#### → Tip

The `serviceKey` payload provides you with the following properties that you will later need for your system configuration:

- `uri` = URL
- `uaa.url` = OAuth2TokenServiceURL
- `uaa.clientid` = User
- `clientsecret` = Password

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

As part of SAP's data model and integration unification strategy, SAP BTP Integration Suite has [Master Data Integration](#) to enable a harmonized integration and distribution of different master data objects and data between SAP solutions. This includes master data for business partners, cost centers, and workforce data. Workforce data is provided for integration scenarios that need data from SAP SuccessFactors Employee Central or other core HR systems.

To learn more, see: [Integrating SAP SuccessFactors Employee Central with SAP Master Data Integration](#)

You can use MDI as a proxy connector to execute *hybrid* scenarios. That means, it can provision its entities via API requests sent by an external back-end system (such as SAP Identity Management or SAP IAG).

**! Restriction**

This scenario supports only executing *read* (GET) operations from MDI to the external back-end system.

**Procedure**

- 1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

**i Note**

If you have a bundle tenant, then in the cockpit → *Neo* → *Overview*, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

- 2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP Master Data Integration](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>



Property Name	Description & Value
URL	Enter the URL to the relevant integration application running in the relevant region of SAP BTP.  See the <b>Prerequisites</b> section → <a href="#">serviceKey</a> tip.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the technical user that has access to the API of your MDI service.  See the <b>Prerequisites</b> section → <a href="#">serviceKey</a> tip.
Password	Enter the password for this technical user.  See the <b>Prerequisites</b> section → <a href="#">serviceKey</a> tip.
OAuth2TokenServiceURL	Enter the OAuth 2.0 Token Service URL.  See the <b>Prerequisites</b> section → <a href="#">serviceKey</a> tip.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP Master Data Integration](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your MDI proxy system. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Field Mapping Between Employee Central and SAP Master Data Integration](#)

Default read and write transformations:

### → Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints ([/Users](#) or [/Groups](#)) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external

application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem",
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$.externalId",
        "targetPath":
"$.externalId",
        "optional": true
      },
      {
        "sourcePath":
"$.profileDetail[0].content.script
edProfileDetails[0].firstName",
        "targetPath":
"$.name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$.profileDetail[0].content.script
edProfileDetails[0].lastName",
        "targetPath":
"$.name.familyName",
        "optional": true
      },
      {
        "sourcePath":
"$.profileDetail[0].content.script
edProfileDetails[0].middleName",
        "targetPath":
"$.name.middleName",
        "optional": true
      },
      {
        "sourcePath":
"$.userAccount.userName",
        "targetPath":
"$.userName",
        "optional": true,
        "correlationAttribute":
true
      },
      {
        "constant": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.emails[?
(@.isDefault == true)].address",

```

## Code Syntax

```

\\ Proxy Write transformation is
currently not supported!

```

```

        "optional": true,
        "correlationAttribute":
true
      },
      {
        "sourcePath": "$.emails",

"preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$.emails"
      },
      {
        "targetPath":
"$ .emails[*].usage",
        "type": "remove"
      },
      {
        "targetPath":
"$ .emails[*].address",
        "type": "rename",
        "constant": "value"
      },
      {
        "targetPath":
"$ .emails[*].isDefault",
        "type": "rename",
        "constant": "primary"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:User",
        "targetPath":
"$ .schemas[0]"
      }
    ]
  }
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[SAP Community: The New Master Data Integration Service for SAP SuccessFactors](#) 

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.28 SAP S/4HANA Cloud

Follow this procedure to set up SAP S/4HANA Cloud as a proxy system.

### Prerequisites

- You have user credentials for an external back-end system with read and write permissions.
- To establish the connection between Identity Provisioning and SAP S/4HANA Cloud, you need to set up the communication (user, system and arrangement) on SAP S/4HANA Cloud. You can do it now (as a prerequisite) or in the process of configuring SAP S/4HANA Cloud as a target system, as described in step 5.

- **i Note**

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context


You can use SAP S/4HANA Cloud as a proxy connector to execute [hybrid](#) scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to SAP S/4HANA Cloud, whenever the external back-end requests such.

In your SAP S/4HANA Cloud system, the HCM (HR) integration is active and cannot be switched off. You have **business users** (Employee, Contingent Worker) and **login users** assigned to them. The corresponding HR integration manages business users – it enables you to update these users from your external data source, such as an identity management system. The Identity Provisioning service manages only the user-related login information, such as date/time preferences or role assignments.

This scenario supports:

- Reading of **users** and **Business roles** (which are considered as [groups](#))
- Writing of **users** and **role assignments**

#### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.

- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

### ❖ Example

```
{
  "sourcePath": "$.user.userName",
  "targetPath": "$.userName",
  "optional": true,
  "correlationAttribute": true
},
```

Since [SAP S/4HANA Cloud](#) doesn't support user filtering, then:

If, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "JOHNSMITH003"**

The query request to the SAP S/4HANA Cloud API will result into a search for a user whose username is 'JOHNSMITH003'.

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [SAP S/4HANA Cloud](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Set up the communication between Identity Provisioning and SAP S/4HANA Cloud and configure your authentication method (basic or certificate-based).

### Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP S/4HANA Cloud proxy system, select the [Certificate](#) tab and choose ► [Generate](#) ► [Download](#) ►, as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip step **a.** if you want to use basic authentication.

The next steps are performed in SAP S/4HANA backend system and are relevant for both basic and certificate-based authentication.

- b. [Create a communication user](#) and provide the respective credentials.

For basic authentication, provide [User Name](#) and [Password](#).



For certificate-based authentication, upload the certificate you have generated in the Identity Provisioning UI on the previous step.

- c. [Create a communication system](#) and assign the created user to the communication system.

For your Identity Provisioning scenario, provide [System ID](#), [System Name](#) and [Host Name](#).

- d. [Create a communication arrangement](#) with the created system.

For your Identity Provisioning scenario, choose [Scenario ID](#) SAP\_COM\_0193 (SAP Cloud Identity Provisioning Integration).

### i Note

The communication scenario [SAP\\_COM\\_0193](#) is enhanced to support the User UUID attribute which is generated by Identity Authentication at user creation.

The User UUID is universally unique identifier. This attribute is immutable and unique across technology layers, such as user interface, APIs, and security tokens, as well as across products and lines of business contributing to a business process in the Intelligent Enterprise.

6. Choose the [Properties](#) tab to configure the connection settings for your system.

### i Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.



If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Enter the SAP S/4HANA Cloud API URL.</p> <p>You can find the correct URL in the <a href="#">API-URL</a> field of the communication arrangement set up for communication scenario SAP_COM_0193.</p> <p>For example: <code>https://my123456-api.s4hana.ondemand.com</code></p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"><li>• <a href="#">BasicAuthentication</a></li><li>• <a href="#">ClientCertificateAuthentication</a></li></ul>

Property Name	Description & Value
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">User Name</a> from the communication arrangement.</p> <div> <b>! Restriction</b>            Do not use special symbol ',' (comma) as it is not supported.         </div>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">Password</a> for the user name from the communication arrangement.</p> <div> <b>! Restriction</b>            Do not use special symbol ',' (comma) as it is not supported.         </div>
s4hana.cloud.api.version	<p>The version of the system API you use.</p> <p>Version <a href="#">1</a> means your SAP S/4HANA Cloud system uses <a href="#">SAP_COM_0193</a> communication arrangement.</p>
s4hana.cloud.skip.read.archived	<p>In the event of archived (disabled) entities in your SAP S/4HANA Cloud system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
s4hana.cloud.hr.switch.active	<p>A default property, whose only possible value is <b>true</b>. That means, HR integration is enabled for your system.</p> <div> <b>⚠ Caution</b>            Do not change this value! Otherwise, your provisioning job will fail.         </div>

Property Name	Description & Value
<code>s4hana.cloud.hr.switch.dependent.role.codes</code>	<p>A default property.</p> <p>As a comma-separated value, add the codes of the roles maintained by the HR integration. Make sure these role codes are part of your <a href="#">read</a> and <a href="#">write</a> transformations.</p> <p>By default, the following codes are added to your system: <b>BUPO03, BBPO05</b>. That means, your HR integration will support <a href="#">employees</a> and <a href="#">contingent worker</a>.</p>
<code>ips.date.variable.format</code>	<p><a href="#">yyyy-MM-dd</a></p> <p>(needed for the read transformation)</p>
<code>s4hana.cloud.user.roles.override</code>	<p>This property defines whether the current roles of a user to be preserved or overwritten by the Identity Provisioning service within the SAP S/4HANA Cloud proxy system in a hybrid scenario.</p> <ul style="list-style-type: none"> <li>• <b>true</b> – the current user roles will be deleted in the proxy system, and the user will be updated only with the roles provisioned by the service.</li> <li>• <b>false</b> – the current user roles will be preserved, and the new roles (if any) will be added for the relevant user in the proxy system.</li> </ul> <p>See also: <a href="#">Extended Explanation of the *user.roles.override Properties</a> </p>
(Optional) <code>s4hana.cloud.roles.filter</code>	<p>Enter OData filtering for reading roles in the S/4HANA system.</p> <p>To learn what criteria you can use, see: <a href="#">OData URI Conventions</a>  → <a href="#">4.5 Filter System Query Option</a></p>
(Optional) <code>s4hana.cloud.roles.page.size</code>	<p>Indicate how many business roles (considered as <a href="#">groups</a>) per page to be read from your SAP S/4HANA Cloud system.</p> <p>The value must be an integer number.</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://my1234567-api.s4hana.ondemand.com
User=MyS4HANAUser
Password=*****
s4hana.cloud.api.version=1
ips.date.variable.format=yyyy-MM-dd
s4hana.skip.read.archived=true
s4hana.onprem.hr.switch.active=true
s4hana.cloud.roles.filter=startswith(ID, 'EMPLOYEE_LEVEL_3') eq true
s4hana.cloud.user.roles.override = false
s4hana.onprem.hr.switch.dependent.role.codes=BP003,BBP010,BBP005
s4hana.cloud.roles.page.size=30
```

---

7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP S/4HANA Cloud](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP S/4HANA Cloud. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP S/4HANA Cloud API: Business User](#)

Default read and write transformations:

→ Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$$.personID",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$$.personalInformation.firstName",
        "targetPath":
"$$.name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$$.personalInformation.lastName",
        "targetPath":
"$$.name.familyName",
        "optional": true
      },
      {
        "sourcePath":
"$$.personalInformation.middleName",
        "targetPath":
"$$.name.middleName",
        "optional": true
      },
      {
        "sourcePath":
"$$.personalInformation.personFullN
ame",
        "targetPath":
"$$.name.formatted",
        "optional": true
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true,
        "targetPath":
"$$.personExternalID"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
        "optional": true,
        "targetPath":
"$$.user.globalUserID"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$$.personID"
      },
      {
        "targetPath":
"$$.businessPartnerRoleCode",
        "type": "valueMapping",
        "sourcePaths": [
          "$$.userType"
        ],
        "defaultValue": "BUP003",
        "valueMappings": [
          {
            "key": [
              "Employee"
            ],
            "mappedValue":
"BUP003"
          },
          {
            "key": [
              "Contingent Worker"
            ],
            "mappedValue":
"BBP005"
          }
        ]
      },
      {
        "scope": "createEntity",

```

```

        "sourcePath":
"$ .user.userName",
        "targetPath":
"$ .userName",
        "optional": true,
        "correlationAttribute":
true
    },
    {
        "constant": true,
        "targetPath": "$ .active"
    },
    {
        "condition":
"$ .user.lockedIndicator ==
'true'",
        "constant": false,
        "targetPath": "$ .active",
        "optional": true
    },
    {
        "condition":
"($ .user.validityPeriod.startDate
> '{currentDate}') ||
('{currentDate}' >
$.user.validityPeriod.endDate)",
        "constant": false,
        "optional": true,
        "targetPath": "$ .active"
    },
    {
        "sourcePath":
"$ .workplaceInformation.emailAddress",
        "targetPath":
"$ .emails[0].value",
        "optional": true,
        "correlationAttribute":
true
    },
    {
        "sourcePath":
"$ .user.logonLanguageCode",
        "optional": true,
        "targetPath": "$ .locale"
    },
    {
        "sourcePath":
"$ .PersonExternalID",
        "optional": true,
        "correlationAttribute":
true
    },
    {
        "sourcePath":
"$ .user.globalUserID",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ][ 'userUid' ]"
    },

```

```

        "sourceVariable":
"currentDate",
        "targetPath":
"$ .user.validityPeriod.startDate"
    },
    {
        "scope": "createEntity",
        "constant": "9999-12-31",
        "targetPath":
"$ .user.validityPeriod.endDate"
    },
    {
        "sourcePath":
"$ .name.givenName",
        "optional": true,
        "targetPath":
"$ .personalInformation.firstName"
    },
    // The following conditions refer
    // to HR integration for your SAP
    // S/4HANA Cloud system. If HR
    // integration is activated
    // (i.e. property
    // s4hana.cloud.hr.switch.active is
    // set to true), then you don't need
    // to provide family name for the
    // users.
    // If it's
    // deactivated (i.e. property
    // s4hana.cloud.hr.switch.active is
    // missing or set to false, then you
    // have to provide family name.
    // You can apply these conditions
    // to different user attributes,
    // analogically to name.familyName.
    {
        "condition":
"%s4hana.cloud.hr.switch.active% !
= null &&
%s4hana.cloud.hr.switch.active%
== true",
        "optional": true,
        "sourcePath":
"$ .name.familyName",
        "targetPath":
"$ .personalInformation.lastName"
    },
    {
        "condition":
"%s4hana.cloud.hr.switch.active%
== null ||
%s4hana.cloud.hr.switch.active%
== false",
        "sourcePath":
"$ .name.familyName",
        "targetPath":
"$ .personalInformation.lastName"
    },
    {
        "sourcePath":
"$ .name.middleName",
        "optional": true,

```

```

    "type": "valueMapping",
    "sourcePaths": [
      "$.user.timeZoneCode"
    ],
    "targetPath":
    "$.timezone",
    "defaultValue": "Europe/
Berlin",
    "valueMappings": [
      {
        "key": [
          "WDFD"
        ],
        "mappedValue":
        "Europe/Berlin"
      },
      {
        "key": [
          "ISRAEL"
        ],
        "mappedValue": "Asia/
Jerusalem"
      },
      {
        "key": [
          "RUS03"
        ],
        "mappedValue":
        "Europe/Moscow"
      },
      {
        "key": [
          "AUSNSW"
        ],
        "mappedValue":
        "Australia/Sydney"
      },
      {
        "key": [
          "UTC+4"
        ],
        "mappedValue": "Asia/
Dubai"
      },
      {
        "key": [
          "BRAZIL"
        ],
        "mappedValue":
        "America/Sao_Paulo"
      },
      {
        "key": [
          "BRZLEA"
        ],
        "mappedValue":
        "America/Sao_Paulo"
      },
      {
        "key": [
          "MSTNO"
        ],

```

```

    "targetPath":
    "$.personalInformation.middleName"
  },
  {
    "sourcePath":
    "$.name.formatted",
    "optional": true,
    "targetPath":
    "$.personalInformation.personFullN
ame"
  },
  {
    "sourcePath":
    "$.userName",
    "targetPath":
    "$.user.userName"
  },
  {
    "sourcePath": "$.locale",
    "optional": true,
    "targetPath":
    "$.user.logonLanguageCode"
  },
  {
    "sourcePath":
    "$.emails[0].value",
    "optional": true,
    "targetPath":
    "$.workplaceInformation.emailAddre
ss"
  },
  {
    "condition": "$.active ==
false",
    "constant": "true",
    "targetPath":
    "$.user.lockedIndicator"
  },
  {
    "scimEntityEndpoint": "Users"
  },
  "group": {
    "mappings": [
      {
        "sourcePath":
        "$.displayName",
        "targetVariable":
        "entityIdTargetSystem",
        "scope": "createEntity"
      },
      {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.displayName"
      },
      {
        "sourcePath":
        "$.members[*].value",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,

```



```

      "mappedValue":
"America/Phoenix"
    },
    {
      "key": [
        "EST"
      ],
      "mappedValue":
"America/New_York"
    },
    {
      "key": [
        "UTC"
      ],
      "mappedValue": "Etc/
UTC"
    },
    {
      "key": [
        "UTC+3"
      ],
      "mappedValue": "Asia/
Riyadh"
    },
    {
      "key": [
        "EST_"
      ],
      "mappedValue":
"America/Toronto"
    },
    {
      "key": [
        "UTC+8"
      ],
      "mappedValue": "Asia/
Shanghai"
    },
    {
      "key": [
        "JAPAN"
      ],
      "mappedValue": "Asia/
Tokyo"
    }
  ],
  {
    "type": "valueMapping",
    "sourcePaths": [
      "$.businessPartnerRoleCode"
    ],
    "targetPath":
"$ .userType",
    "defaultValue":
"Employee",
    "valueMappings": [
      {
        "key": [
          "BUP003"
        ],

```

```

      "targetPath": "$.members[?
(@.value)]"
    },
    {
      "sourcePath":
"$ .Operations",
      "targetPath":
"$ .Operations",
      "preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    },
    {
      "sourcePath": "$.schemas",
      "targetPath": "$.schemas",
      "preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    }
  ],
  "scimEntityEndpoint": "Groups"
}

```

```

      "mappedValue":
"Employee"
      },
      {
        "key": [
          "BBP005"
        ],
        "mappedValue":
"Contingent Worker"
      }
    ]
  },
  {
    "sourcePath":
"$ .user.role[*].roleName",
    "preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath": "$ .groups[?
(@.value)]"
  }
]
},
"group": {
  "scimEntityEndpoint":
"Groups",
  "mappings": [
    {
      "sourcePath": "$ .ID",
      "targetPath": "$ .id",
      "targetVariable":
"entityIdSourceSystem"
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetVariable":
"entityLocationSourceSystem",
      "targetPath":
"$ .meta.location",
      "functions": [
        {
          "type":
"concatString",
          "suffix": "$
{entityIdSourceSystem}"
        }
      ]
    },
    {
      "sourcePath": "$ .ID",
      "targetPath":
"$ .displayName"
    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
      "targetPath":
"$ .schemas[0]"
    }
  ],

```

```

    {
      "sourcePath":
"$$.to_BusinessUserAssignment.results",
      "optional": true,

      "preserveArrayWithSingleElement":
true,
      "targetPath": "$.members"
    },
    {
      "type": "remove",
      "targetPath":
"$$.members[*].__metadata"
    },
    {
      "type": "rename",
      "constant": "value",
      "targetPath":
"$$.members[*].PersonID"
    },
    {
      "constant": "User",
      "targetPath":
"$$.members[*].type"
    }
  ]
}

```

By default, Identity Provisioning reads group IDs and members. If you want the service to also read group descriptions, you can add an extra mapping to the *"group"* resource in the [Read Transformation](#). To learn how, see [Guided Answers: Business Role Description](#).

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PATCH**.

### **i Note**

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## **Next Steps**

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### **⚠ Caution**

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## **Related Information**

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.29 SAP S/4HANA for procurement planning

Follow this procedure to set up SAP S/4HANA for procurement planning as a proxy system.

### Prerequisites

#### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

You have technical credentials for SAP S/4HANA for procurement planning. See: [Onboarding](#)

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

SAP S/4HANA for procurement planning is a cloud-based solution designed to help you plan procurement activities with regard to the time schedule, as well as the investment planning of items based on a central bill of material.

You can use Identity Provisioning to configure SAP S/4HANA for procurement planning as a proxy system in hybrid scenarios. For example, when SAP S/4HANA for procurement planning is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management, without making a direct connection between both systems. You can provision users to the external backend system, which can trigger CRUD (create, read, update, delete) operations on users back to the SAP S/4HANA for procurement planning.

This scenario supports provisioning **users**.

#### i Note

SAP S/4HANA for procurement planning does not support groups.

### Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit ► [Neo](#) ► [Overview](#) ►, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li> <li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li> <li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li> <li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP S/4HANA Procurement Planning* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.


We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL to the SCIM API of your SAP S/4HANA for procurement planning system without path information.</p> <p>For example: <code>https://procplanning-api.cfapps.eu10.hana.ondemand.com</code></p>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the OAuth Client Id, created for your SAP S/4HANA for procurement planning system.
Password	Enter the OAuth Client Secret, created for your SAP S/4HANA for procurement planning system.
OAuth2TokenServiceURL	<p>Enter the OAuth 2.0 Token Service URL.</p> <p>For example: <b><code>https://procplansecurity.authentication.eu10.hana.ondemand.com/oauth/token</code></b></p>
(Optional) <code>s4hana.pp.user.filter</code>	<p>When specified, only those SAP S/4HANA for procurement planning users matching the filter expression will be read.</p> <p>Example: <b><code>name.familyName eq "Smith" and addresses.country eq "US"</code></b></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example: [GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `scim.user.filter = timezone eq "Africa"`

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP S/4HANA for procurement planning API will result into: **/Users?filter=timezone eq "Africa" and userName eq "johnsmith03"**

#### 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP S/4HANA for procurement planning Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP S/4HANA for procurement planning system. For more information, see:

[Manage Transformations \[page 1494\]](#).

[SAP Business Accelerator Hub: SAP S/4HANA for Procurement Planning](#)

Default read and write transformations:



### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users* or */Groups*) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourcePath":
"$$.userName",
        "correlationAttribute": true,
        "targetPath":
"$$.userName"
      },
      {
        "sourcePath":
"$$.emails[0].value",
        "targetPath":
"$$.emails[0].value",
        "optional": true
      },
      {
        "sourcePath":
"$$.emails[?(@.primary==
true)].value",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
      },
      {
        "sourcePath":
"$$.active",
        "targetPath":
"$$.active"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName"
      },
      {
        "sourcePath":
"$$.displayName",
        "targetPath":
"$$.displayName"
      },
      {
        "sourcePath":
"$$.active",
        "optional": true,
        "targetPath":
"$$.active"
      },
      {
        "sourcePath":
"$$.externalId",
        "optional": true,
        "targetPath":
"$$.externalId"
      },
      {
        "sourcePath":
"$$.emails",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.emails"
      }
    ]
  }
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be

automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"><li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li><li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li></ul>	<ul style="list-style-type: none"><li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li><li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li></ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[SAP S/4HANA for procurement planning – Product Page](#)

## 1.6.3.30 SAP S/4HANA On-Premise

Follow this procedure to set up SAP S/4HANA on-premise (also valid for SAP S/4HANA Cloud, private edition) as a proxy system.

### Prerequisites

#### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)
  - You have technical credentials (user and password) for an external back-end system with read and write permissions.
  - You have technical credentials (user and password) for SAP S/4HANA on-premise.
  - The SAP S/4HANA on-premise system is version **1809** or higher.
  - You have configured your SOA Manager to directly call the following Web services:
    - **ManageBusinessUserIn**
    - **QueryBusinessUserIn**
- For more information, see: [Setting Up SOA Manager](#).

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

You can use SAP S/4HANA on-premise as a proxy connector to execute *hybrid* scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to as a proxy connector to execute SAP S/4HANA on-premise, whenever the external back-end requests such. This scenario supports:

- Reading of **users**
- Writing of **users**


SAP S/4HANA on-premise supports provisioning of users with *User UUID* attribute which is generated by Identity Authentication at user creation. The attribute mapping is handled by the default transformation of AS ABAP. Therefore, to provision a user with *User UUID* to or from S/4HANA on-premise, the user should first be provisioned to AS ABAP and then linked to its corresponding business user in S/4HANA on-premise. For more information, see: [SAP Application Server ABAP \[page 1059\]](#).

According to your use case, you can decide whether to use SAP S/4HANA on-premise with HR (human resources) integration active or not.

- System with HR integration – When the HR integration is active, business users in SAP S/4HANA on-premise are created and managed by the HR system. The Identity Provisioning service can manage only the user-related login information, such as date/time preferences, or role assignments. This means that once the business users are created in SAP S/4HANA on-premise, you need to provision the users to AS ABAP so that they are linked to the business users (employee, collaboration user, contingent worker, resource ) via the [User\\_Assignment](#) attribute.
- System without HR integration – lean business users will be created after the provisioning job, and the AS ABAP users will be linked to them. The Identity Provisioning service manages the complete set of **business partners** and their relevant **business users**.

To enable HR integration, you need to specify the relevant property in the system configuration. See **step 7** from the procedure below.

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

#### Example

```
{
  "sourcePath": "$.userAssignment.userID",
  "targetPath": "$.userName",
  "optional": true,
  "correlationAttribute": true
},
```

Since *SAP S/4HANA On-Premise* doesn't support user filtering, then:

If, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=username eq "JOHNSMITH003"**

The query request to the SAP S/4HANA On-Premise API will result into a search for a user whose username is 'JOHNSMITH003'.

## Procedure

1. Open the Cloud Connector to add an access control system mapping for **SAP S/4HANA On-Premise**. This is needed to allow the Identity Provisioning service to access SAP S/4HANA On-Premise as a back-end system on the intranet. To learn how, see: [Configure Access Control \(HTTP\)](#)
2. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### Note

If you have a bundle tenant, then in the cockpit → *Neo* → *Overview*, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

3. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

#### SAP Cloud Identity Infrastructure

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

#### Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

4. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
5. Add [SAP S/4HANA On-Premise](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
6. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to your SAP S/4HANA On-Premise system.

Property Name	Description & Value
ProxyType	Enter: <a href="#">OnPremise</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the technical user for SAP S/4HANA On-Premise.
Password	(Credential) Enter the password for the SAP S/4HANA On-Premise technical user.
<code>s4hana.onprem.skip.read.archived</code>	<p>In the event of archived (disabled) entities in a source SAP S/4HANA On-Premise system, choose whether the provisioning jobs to continue reading such entities or to skip them.</p> <p>This property is enabled by default. If you want to always read disabled entities, set the property to <b>false</b>, or delete it.</p>
<code>s4hana.onprem.hr.switch.active</code>	<p>This property is disabled by default. Possible values:</p> <ul style="list-style-type: none"> <li>• <b>true</b> – HR integration is enabled for your system</li> <li>• <b>false</b> (default value) – HR integration is disabled for your system</li> </ul>
<code>s4hana.onprem.hr.switch.dependent.role.codes</code>	<p>A default property. Relevant only for systems with activated HR integration, that is if <code>s4hana.onprem.hr.switch.active = true</code>.</p> <p>As a comma-separated value, add the codes of the roles maintained by the HR integration. Make sure these role codes are part of your <a href="#">read</a> and <a href="#">write</a> transformations.</p> <p>By default, the following codes are added to your system: <b> BUP003, BBP005, BUP012, WFM001</b>. That means, your HR integration will support <a href="#">employees</a>, <a href="#">contingent worker</a>, <a href="#">collaboration user</a>, and <a href="#">resource</a>.</p>



Property Name	Description & Value
<code>s4hana.onprem.user.unique.attribute</code>	<p>If Identity Provisioning tries to provision a user that already exists in the SAP S/4HANA On-Premise target system (a conflicting user), this property defines the unique attributes by which the existing user will be searched and resolved.</p> <p>According to your use case, choose how to set up this property:</p> <ul style="list-style-type: none"> <li>Default behavior: This property does not appear in the UI during system creation. Its default value is <code>personExternalID</code>. That means, if the service finds an existing user by a <code>personExternalID</code>, it updates this user with the data of the conflicting one. If a user with such a <code>personExternalID</code> is not found, the creation of the conflicting user fails.</li> <li>Value = <code>emails[0].value</code>. If the service finds an existing user matching both unique attributes <code>email</code> and <code>personExternalID</code>, it updates this user with the data of the conflicting one. If the service finds an existing user matching only the <code>email</code>, the update of the existing user fails. If a user with such <code>email</code> is not found, that means the conflict is due to another reason, so the creation of the conflicting user fails.</li> </ul> <p>Possible values:</p> <ul style="list-style-type: none"> <li><code>personExternalID</code></li> <li><code>emails[0].value</code></li> </ul> <p>Default value: <code>personExternalID</code></p>
<code>ips.date.variable.format</code>	<p><code>yyyy-MM-dd</code></p> <p>(needed for the read transformation)</p>
<p>(Optional)</p> <p><code>s4hana.onprem.sap-client</code></p>	<p>Use this property if you want to specify a particular AS ABAP client to use as the <b>sap-client</b> URL parameter.</p> <p>If this property is not specified, the URL will open your default AS ABAP client. To learn more, see: <a href="#">Specifying the Client</a></p> <p>For more information about <b>sap-client</b>, see: <a href="#">SAP URL Parameters</a></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=OnPremise
URL=http://aaa777.myhost:1234
User=MYS4HANAUSER
Password=*****
ips.date.variable.format=yyyy-MM-dd
s4hana.onprem.skip.read.archived=true
s4hana.onprem.hr.switch.active=true
s4hana.onprem.hr.switch.dependent.role.codes = BUP003,BBP010,BBP005
s4hana.onprem.sap-client=101
```

---

7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP S/4HANA On-Premise](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP S/4HANA On-Premise. For more information, see:

[Manage Transformations \[page 1494\]](#)

[APIs for Business User Management](#)

Default read and write transformations:

→ Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy [Write Transformation](#) is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a [target](#) one.

However, after a [Create](#) or [Update](#) operation is performed on the proxy system, the [Read Transformation](#) is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy [Read Transformation](#) is used for [write](#) cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$ .personID",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$ .id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$ .personalInformation.firstName",
        "targetPath":
"$ .name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$ .personalInformation.lastName",
        "targetPath":
"$ .name.familyName",
        "optional": true
      },
      {
        "sourcePath":
"$ .personalInformation.middleName",
        "targetPath":
"$ .name.middleName",
        "optional": true
      },
      {
        "sourcePath":
"$ .personalInformation.personFullN
ame",
        "targetPath":
"$ .name.formatted",
        "optional": true
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .personExternalID"
      },
      {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
        "optional": true,
        "targetPath":
"$ .personExternalID"
      },
      {
        "targetPath":
"$ .personID",
        "sourceVariable":
"entityIdTargetSystem"
      },
      {
        "type": "valueMapping",
        "sourcePaths": [
          "$ .userType"
        ],
        "targetPath":
"$ .businessPartnerRoleCode",
        "defaultValue": "BUP003",
        "valueMappings": [
          {
            "key": [
              "Employee"
            ],
            "mappedValue":
"BUP003"
          },
          {
            "key": [
              "Contingent Worker"
            ],
            "mappedValue":
"BBP005"
          },
          {
            "key": [
              "Collaboration User"
            ],
            "mappedValue":
"BUP012"
          },
          {
            "key": [
              "Resource"
            ]
          }
        ]
      }
    ]
  }
}

```

```

      "sourcePath":
"$ .personalInformation.nickName",
      "targetPath":
"$ .nickName",
      "optional": true
    },
    {
      "sourcePath":
"$ .userAssignment.userID",
      "targetPath":
"$ .userName",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "constant": true,
      "targetPath": "$ .active"
    },
    {
      "sourcePath":
"$ .workplaceInformation.emailAddress",
      "targetPath":
"$ .emails[0].value",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .PersonExternalID",
      "optional": true,
      "correlationAttribute":
true
    },
    {
      "type": "valueMapping",
      "sourcePaths": [
"$ .businessPartnerRoleCode"
],
      "targetPath":
"$ .userType",
      "defaultValue":
"Employee",
      "valueMappings": [
        {
          "key": [
            "BUP003"
          ],
          "mappedValue":
"Employee"
        },
        {
          "key": [
            "BBP005"
          ],
          "mappedValue":
"Contingent Worker"
        }
      ]
    }
  ],

```

```

      "mappedValue":
"WFM001"
    }
  ],
  {
    "scope": "createEntity",
    "targetPath":
"$ .validityPeriod.startDate",
    "sourceVariable":
"currentDate"
  },
  {
    "scope": "createEntity",
    "targetPath":
"$ .validityPeriod.endDate",
    "constant": "9999-12-31"
  },
  {
    "sourcePath":
"$ .name.givenName",
    "targetPath":
"$ .personalInformation.firstName",
    "optional": true
  },
  {
    "condition":
"%s4hana.onprem.hr.switch.active%
!= null &&
%s4hana.onprem.hr.switch.active%
== true",
    "optional": true,
    "sourcePath":
"$ .name.familyName",
    "targetPath":
"$ .personalInformation.lastName"
  },
  {
    "condition":
"%s4hana.onprem.hr.switch.active%
== null ||
%s4hana.onprem.hr.switch.active%
== false",
    "sourcePath":
"$ .name.familyName",
    "targetPath":
"$ .personalInformation.lastName"
  },
  {
    "sourcePath":
"$ .name.middleName",
    "targetPath":
"$ .personalInformation.middleName"
  },
  {
    "optional": true
  },
  {
    "sourcePath":
"$ .name.formatted",
    "targetPath":
"$ .personalInformation.personFullName",
    "optional": true
  }

```

## Read Transformation

```

    "key": [
      "BUP012"
    ],
    "mappedValue":
"Collaboration User"
  },
  {
    "key": [
      "WFM001"
    ],
    "mappedValue":
"Resource"
  }
]
}

```

## Write Transformation

```

    },
    {
      "sourcePath":
"$nickName",
      "targetPath":
"$personalInformation.nickName",
      "optional": true
    },
    {
      "sourcePath":
"$userName",
      "targetPath":
"$userAssignment.userID",
      "optional": true,
      "functions": [
        {
          "function":
"toUpperCaseString"
        }
      ]
    },
    {
      "sourcePath":
"$emails[0].value",
      "targetPath":
"$workplaceInformation.emailAddress",
      "optional": true
    }
  ]
}

```

8. (Optional) If the external consumer system is **SAP Identity Management (IDM)**, you can export the newly created proxy system as a **.csv** file. This will reduce the effort of manually entering all the properties again when an IDM administrator import the proxy configuration as a repository.

To export the proxy system from the UI, choose **Export > CSV format**.

### → Tip

When you import the **.csv** file as a SCIM repository in IDM, all the property fields will be automatically filled-in. However:

- You have to manually enter your client ID and secret (AUTH\_USER and AUTH\_PASSWORD).
- As currently the scenario does not support groups, you cannot make group assignments for users. Thus, for the SCIM\_ASSIGNMENT\_METHOD constant, remove the value (leave the field empty).

### i Note

If you have purchased the Identity Provisioning service after **September 1, 2020**, for AUTH\_USER and AUTH\_PASSWORD, enter the technical user and password for which you have granted access to the Identity Provisioning Proxy API.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[SAP S/4HANA On-Premise](#)

[APIs for Business User Management](#)

[Maintain Collaboration Users](#)

## 1.6.3.31 SAP Sales Cloud and SAP Service Cloud

Follow this procedure to set up SAP Sales Cloud and SAP Service Cloud, formerly known as [SAP Cloud for Customer](#) (in short, C4C), as a proxy system.


## Prerequisites

### ! Restriction

This system is available for bundle tenants running on SAP Cloud Identity infrastructure and standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants

running on Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

To integrate SAP Sales Cloud and SAP Service Cloud with Identity Provisioning, you need to use SAP Cloud Integration (SAP CI). This service provides a package with integration flows (iFlows) for enabling the creation of users and assignment of users to groups via SCIM API in SAP Sales Cloud and SAP Service Cloud.

- To configure SAP Cloud Integration and SAP Sales Cloud and SAP Service Cloud, see: [Identity Provisioning in SAP Cloud for Customer using System for Cross-Domain Identity Management \(SCIM\)](#)
- To set up and use the *SAP Cloud for Customer Integration with Identity Provisioning via System for Cross-domain Identity Management* package, see: [SAP Business Accelerator Hub: SAP Cloud for Customer Integration with Identity Provisioning via System for Cross-domain Identity Management](#) 

### **i Note**

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## **Context**

SAP Sales Cloud and SAP Service Cloud is a cloud-based solution that helps customers manage day-to-day sales and service interactions by sending and receiving signals between front- and back-office solutions and providing a single view of the customer.

You can use Identity Provisioning to configure SAP Sales Cloud and SAP Service Cloud as a proxy system for integration with on-premise or cloud identity management systems that support SCIM 2.0 standard (for example, SAP Identity Management and SAP Cloud Identity Access Governance).

In this scenario Identity Provisioning acts as a proxy between SAP Sales Cloud and SAP Service Cloud and the on-premise or cloud system. Identity Provisioning exposes SAP Sales Cloud and SAP Service Cloud as a proxy system and connects it to the identity management system without making a direct connection between both systems. You can then provision users to the identity management system, which can trigger CRUD (create, read, update, delete) operations on users back to the SAP Sales Cloud and SAP Service Cloud.

## **Procedure**

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### **i Note**

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <i>Users &amp; Authorizations</i> ► <i>Administrators</i> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <i>Access Proxy System API</i> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <i>Security</i> ► <i>OAuth</i> ► <i>Clients</i> ► and choose <i>Register New Client</i>.</li> <li>2. From the <i>Subscription</i> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <i>Authorization Grant</i> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <i>Secret</i> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <i>Client ID</i>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <i>Subscriptions</i> ► <i>Java Applications</i> ► <i>ipsproxy</i> ►.</li> <li>7. From the left-side navigation, choose ► <i>Roles</i> ► <i>IPS_PROXY_USER</i> ►.</li> <li>8. Choose <i>Assign</i> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP Sales Cloud and SAP Service Cloud* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the *Properties* tab to configure the connection settings for your system.

### **i** Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.




## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter SAP Cloud Integration runtime URL. See: <a href="#">How to Get SAP Cloud Integration Runtime URL</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter SAP Cloud Integration user ID to connect to SAP Cloud Integration. See: <ul style="list-style-type: none"> <li><a href="#">Setting Up Inbound HTTP Connections (with Basic Authentication), Neo Environment</a></li> <li><a href="#">Basic Authentication of IdP User for Integration Flow Processing (Cloud Foundry environment)</a></li> </ul>
Password	(Credential) Enter SAP Cloud Integration password to connect to SAP Cloud Integration. See: <ul style="list-style-type: none"> <li><a href="#">Setting Up Inbound HTTP Connections (with Basic Authentication), Neo Environment</a></li> <li><a href="#">Basic Authentication of IdP User for Integration Flow Processing (Cloud Foundry environment)</a></li> </ul>
c4c.api.version	The version of the SAP Sales Cloud and SAP Service Cloud API you use. By default, the Identity Provisioning service uses version <b>3</b> - the SCIM 2.0 based API.
(Optional) c4c.user.filter	When specified, only those C4C users matching the filter expression will be read.  For example: <ul style="list-style-type: none"> <li>userName eq "Smith"</li> <li>email eq "test@abc.com"</li> <li>employeeNumber eq "56789"</li> <li>addresses.country eq "USA"</li> </ul>
(Optional) c4c.group.filter	When specified, only those C4C groups matching the filter expression will be read.  Example: <b>displayName eq "ProjectTeam1"</b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate

the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the *Read Transformation*, there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used '*eq*' filter).
- Fully qualified names (<*schema*>:<*attribute*>) are not supported. For example:  
*GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the *Properties* tab: *c4c.user.filter = timezone eq "US"*

Then if, for example, the SCIM Proxy endpoint request is: *GET .../Users?filter=userName eq "johnsmith03"*

The query request to the C4C API will result into: */Users?filter=timezone eq "US" and userName eq "johnsmith03"*

## 6. Configure the transformations.

The Identity Provisioning offers a default transformation for the *SAP Sales Cloud and SAP Service Cloud* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in C4C. For more information, see:

[Manage Transformations \[page 1494\]](#)

Default read and write transformations:

→ Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath": "$.name",
        "optional": true,
        "targetPath": "$.name"
      },
      {
        "sourcePath":
"$ .displayName",
        "optional": true,
        "targetPath":
"$ .displayName"
      },
      {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath":
"$ .addresses",
        "targetPath":
"$ .addresses",
        "optional": true,

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "optional": true,
        "targetPath": "$.name"
      },
      {
        "sourcePath": "$.active",
        "optional": true,
        "targetPath": "$.active"
      },
      {
        "sourcePath": "$.title",
        "optional": true,
        "targetPath": "$.title"
      },
      {
        "sourcePath": "$.locale",
        "optional": true,
        "targetPath": "$.locale"
      },
      {
        "sourcePath":
"$ .nickName",
        "optional": true,
        "targetPath": "$.nickName"
      },
      {
        "sourcePath":
"$ .addresses",
        "optional": true,
        "targetPath":
"$ .addresses",
        "preserveArrayWithSingleElement":
true
      },
      {
        "sourcePath":
"$ .phoneNumbers",
        "optional": true,
        "targetPath":
"$ .phoneNumbers",
        "preserveArrayWithSingleElement":
true
      },

```

```

"preserveArrayWithSingleElement":
true
    {
        "sourcePath":
"$ .phoneNumbers",
        "targetPath":
"$ .phoneNumbers",
        "optional": true,

"preserveArrayWithSingleElement":
true
    {
        "sourcePath":
"$ .userType",
        "targetPath":
"$ .userType",
        "optional": true
    },
    {
        "sourcePath":
"$ .nickName",
        "optional": true,
        "targetPath": "$ .nickName"
    },
    {
        "sourcePath":
"$ .preferredLanguage",
        "optional": true,
        "targetPath":
"$ .preferredLanguage"
    },
    {
        "sourcePath": "$ .emails",
        "targetPath": "$ .emails",

"preserveArrayWithSingleElement":
true,
        "optional": true
    },
    {
        "sourcePath": "$ .schemas",
        "targetPath": "$ .schemas",

"preserveArrayWithSingleElement":
true
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]"
    },
    {
        "sourcePath": "$ .groups",

```

```

    {
        "sourcePath":
"$ .userType",
        "optional": true,
        "targetPath": "$ .userType"
    },
    {
        "sourcePath":
"$ .preferredLanguage",
        "optional": true,
        "targetPath":
"$ .preferredLanguage"
    },
    {
        "sourcePath": "$ .emails",
        "targetPath": "$ .emails",

"preserveArrayWithSingleElement":
true,
        "optional": true
    },
    {
        "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]",
        "optional": true,
        "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User' ]
[ 'employeeNumber' ]"
    },
    {
        "sourcePath": "$ .schemas",
        "targetPath": "$ .schemas",

"preserveArrayWithSingleElement":
true
    }
]
},
"group": {
    "skipOperations": [
        "create",
        "delete"
    ],
    "mappings": [
        {
            "sourceVariable":
"entityIdTargetSystem",
            "targetPath": "$ .id"
        },
        {
            "sourcePath":
"$ .displayName",
            "targetPath":
"$ .displayName",
            "optional": true
        },
        {
            "sourcePath":
"$ .Operations",

```

```

"preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath": "$.groups"
  }
],
"group": {
  "scimEntityEndpoint":
"Groups",
  "mappings": [
    {
      "sourcePath": "$.id",
      "targetPath": "$.id",
      "targetVariable":
"entityIdSourceSystem"
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetVariable":
"entityLocationSourceSystem",
      "targetPath":
"$ .meta .location",
      "functions": [
        {
          "type":
"concatString",
          "suffix": "$
{entityIdSourceSystem}"
        }
      ]
    },
    {
      "sourcePath":
"$ .displayName",
      "targetPath":
"$ .displayName"
    },
    {
      "sourcePath": "$.members",
      "optional": true,
      "targetPath": "$.members",
    }
  ],
  "preserveArrayWithSingleElement":
true
    },
    {
      "sourcePath": "$.schemas",
    }
  ],
  "preserveArrayWithSingleElement":
true,
    "targetPath": "$.schemas"
  }
]
}

```

```

    "targetPath":
"$ .Operations",
  "preserveArrayWithSingleElement":
true,
    "scope": "patchEntity"
  },
  {
    "sourcePath": "$.schemas",
    "targetPath": "$.schemas",
  }
],
"preserveArrayWithSingleElement":
true,
    "scope": "patchEntity"
  },
  {
    "sourcePath": "$.members",
  }
],
"preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath": "$.members"
  },
  {
    "sourcePath": "$.schemas",
    "targetPath": "$.schemas",
  }
],
"preserveArrayWithSingleElement":
true
  }
],
"scimEntityEndpoint": "Groups"
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

### 1.6.3.32 SAP SuccessFactors

Follow this procedure to set up SAP SuccessFactors as a proxy system.

#### Prerequisites

- You have created a technical user with permissions to **call** the SAP SuccessFactors HCM Suite OData API and to **export** employee data from the SAP SuccessFactors system. For more information, see [Permissions](#) and [URI Conventions \(OData Version 2.0\)](#) ↗.
- You have the [Admin Center](#) > [Manage Permission Roles](#) > [Access to X.509 Certificates](#) permission (needed for configuring X.509 certificate-based authentication)

#### Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

#### Context

You can use SAP SuccessFactors as a proxy connector to execute *hybrid* scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to SAP SuccessFactors, whenever the external back-end requests such.

SAP SuccessFactors provides two APIs for its integration with Identity Provisioning: SAP SuccessFactors HCM Suite OData API and SAP SuccessFactors Workforce SCIM API. The value of `sf.api.version` property controls which API you use.

- When the value is set to **1**, or the property is not defined - SAP SuccessFactors HCM Suite OData API (in short, OData API) is used. This is the default value. SAP SuccessFactors source systems created before the introduction of `sf.api.version` property, use OData API. This version allows you to read users and groups, both static and dynamic. It also supports writing users, updating dynamic groups and group members.

#### ! Restriction

Note the following restrictions when using SAP SuccessFactors HCM Suite OData API:

- You cannot [create](#) or [delete](#) groups as these operations are currently not supported.
- Managing SAP SuccessFactors **static groups** is not supported.



- When the value is set to **2** - SAP SuccessFactors Workforce SCIM API (in short, SCIM API) is used. This version allows you to provision static permission groups and user's group assignments. Provisioning of user's group assignments from a given external system to SAP SuccessFactors proxy system is possible if the user is an active employee with a work assignment. To update a group from the external system, a group with the same name should already exist in SAP SuccessFactors. For more information about the difference between static and dynamic groups in SAP SuccessFactors, see [Permission Groups](#).

### ! Restriction


Note the following restrictions when using SAP SuccessFactors Workforce SCIM API:

- You cannot [create](#) or [delete](#) groups using the `Groups` APIs.
- The `Groups` patch API only supports updating membership of static permission groups.

For more information, see [Overview of SAP SuccessFactors Workforce System for Cross-Domain Identity Management API](#).

For more information on how to update to version 2, see [Update Connector Version \[page 1484\]](#).

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use '**eq**' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- 0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- 1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.%ldap.attribute.user.id%{0}",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `sf.user.filter = lastName eq "Smith"`

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=id eq "p01234567"**

The query request to the SAP SuccessFactors API will result into: **/User?\$filter=lastName eq 'Smith' and userId eq 'p01234567'**

## Procedure

- Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

- Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.


SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li> <li>Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>Save your changes.</li> <li>Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li> <li>Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li> <li>From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li> <li>In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li> <li>From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li> <li>From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li> <li>Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

- Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
- Add *SAP SuccessFactors* as a proxy system. For more information, see [Add a System \[page 1477\]](#).

5. Set up the communication between Identity Provisioning and SAP SuccessFactors and configure your authentication method (basic or certificate-based).

### Note

We recommend that you use certificate-based authentication.

- a. In your newly added SAP SuccessFactors proxy system, select the [Certificate](#) tab and choose [Generate](#) > [Download](#) , as described in [Generate and Manage Certificates for Outbound Connection](#).

Skip this step if you use basic authentication. The next steps are performed in SAP SuccessFactors Admin Center and are relevant for certificate-based authentication only.

- b. Login to SAP SuccessFactors and go to [Admin Center](#). Follow the procedure described in [Upgrade to X.509 Certificate-Based Authentication for Incoming Calls](#).

Make sure you select [Identity Provisioning Service](#) in the [Integration Name](#) field.

6. Choose the [Properties](#) tab to configure the connection settings for your system.


### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	<p>Specify the URL to your SAP SuccessFactors API.</p> <p>For example:</p> <ul style="list-style-type: none"><li>For version 1: <a href="https://apitest.successfactors.com/odata/v2">https://apitest.successfactors.com/odata/v2</a></li><li>For version 2: <a href="https://apitest.successfactors.com">https://apitest.successfactors.com</a></li></ul> <p>To see the list of all SAP SuccessFactors data centers, see: <a href="#">HXM Suite OData APIs: API Endpoint URLs and System for Cross-domain Identity Management for Workforce in SuccessFactors</a> </p>
ProxyType	Enter: <a href="#">Internet</a>

Property Name	Description & Value
Authentication	<p>Enter your authentication method:</p> <ul style="list-style-type: none"> <li>• <a href="#">BasicAuthentication</a></li> <li>• <a href="#">ClientCertificateAuthentication</a></li> </ul>
(Optional) <code>sf.api.version</code>	<p>Handles the version of the API which is consumed by the SAP SuccessFactors system.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">1</a> - Indicates that SAP SuccessFactors HCM Suite OData API (in short, OData API) is used.</li> <li>• <a href="#">2</a> - Indicates that SAP SuccessFactors Workforce SCIM API (in short, SCIM API) is used.</li> </ul> <p>Default value: <a href="#">1</a></p>
User	<p>Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the <a href="#">userID</a> of your SAP SuccessFactors technical user in the following format: <code>&lt;user_ID&gt;@&lt;company_ID&gt;</code></p>
Password	<p>(Credential) Valid if <a href="#">BasicAuthentication</a> is configured as authentication method.</p> <p>Enter the password for your SAP SuccessFactors technical user.</p>
<code>sf.company.id</code>	<p>Valid if <a href="#">ClientCertificateAuthentication</a> is configured as authentication method.</p> <p>Enter the Company ID of your SAP SuccessFactors system.</p> <p>The Company ID is a short string of characters that identifies each SAP SuccessFactors system. It is like a username for your organization. All users of the same system share the same Company ID.</p>

Property Name	Description & Value
<code>sf.user.attributes</code>	<p>Default property. It's a string representing a comma-separated list of user attributes that have to be loaded (read) from SAP SuccessFactors. You can leave the default property value (all listed attributes), or leave only some of them.</p> <div> <p>→ Remember</p> <ul style="list-style-type: none"> <li>Always make sure that attribute <code>lastModifiedDateTime</code> is in the list of values. If you don't specify it, the provisioning from SAP SuccessFactors will fail.</li> <li>If a user in SAP SuccessFactors is missing this attribute, it will break the provisioning. You can exclude them from the provisioning (either by using the <code>sf.user.filter</code> property, or by setting a condition in the transformation logic).</li> </ul> </div> <p><b>Connector version:</b> SAP SuccessFactors version 1</p>
(Optional) <code>sf.user.attributes.expand</code>	<p>This property reads additional user data related to <a href="#">complex attributes</a>, which are specified in <code>sf.user.attributes</code>.</p> <p>Default value: <a href="#">personKeyNav</a></p> <p><b>Connector version:</b> SAP SuccessFactors version 1</p>
(Optional) <code>sf.user.filter</code>	<p>The possible values of this property depend on the API version which your SAP SuccessFactors system consumes.</p> <p>Use this property to filter users from SAP SuccessFactors. The filter obtains values as described in the OData 2.0 syntax, except any statements with attribute <code>lastModifiedDateTime</code>. To learn more, see:</p> <ul style="list-style-type: none"> <li><a href="#">OData version 2</a> → <b>4.5. Filter System Query Option (\$filter)</b>.</li> <li><a href="#">SAP SuccessFactors HXM Suite OData API: Reference Guide (V2)</a></li> <li><a href="#">SAP SuccessFactors Workforce SCIM API and System for Cross-domain Identity Management for Workforce in SuccessFactors</a></li> </ul>

Property Name	Description & Value
(Optional) <code>sf.user.read.deactivatedafter</code>	<p>This property filters SAP SuccessFactors inactive users from a particular date on. It is an optional property which does not appear by default at system creation. It accepts a value in the <b>yyyy-MM-dd</b> format. For example: <b>2023-07-17</b></p> <p>The <code>sf.user.read.deactivatedafter</code> property works together with the <code>sf.user.filter</code> property which is added at system creation with the default value: <code>active eq true</code>. Using it can further narrow down the filtering results.</p> <p>To filter active users along with inactive ones from a particular date on, the following configuration must be in place:</p> <ul style="list-style-type: none"> <li>Set the <code>sf.user.read.deactivatedafter</code> value to a date in the expected format. For example: <code>2023-07-17</code></li> <li><code>sf.user.filter = active eq true</code></li> </ul> <p>As a result, Identity Provisioning reads SAP SuccessFactors active users and the users set to inactive from that date on using the 2023-07-17T00:00:00Z date-time format.</p> <p>Depending on the value you define for <code>sf.user.filter</code>, expect the following results:</p> <ul style="list-style-type: none"> <li><code>sf.user.filter = active eq false</code> All inactive users will be returned.</li> <li><code>sf.user.filter = active eq false and userName sw "Test_"</code> All inactive users with username starting with Test_ will be returned.</li> </ul> <div> <p><b>i Note</b></p> <p>When you filter by <code>sf.user.filter = active eq false</code> along with the property <code>sf.user.read.deactivatedafter</code>, the users that match the two criteria will be read twice.</p> </div> <ul style="list-style-type: none"> <li><code>sf.user.filter = active eq true and userName sw "Test_"</code> Inactive users from the provided date on and all active users with username starting with Test_ will be returned.</li> </ul>

Property Name	Description & Value
<b>Connector version:</b> SAP SuccessFactors version 2	
(Optional) <code>sf.group.filter</code>	<p>The possible values of this property depend on the API version which your SAP SuccessFactors system consumes.</p> <p>Use this property to filter dynamic groups in SAP SuccessFactors. The filter obtains values as described in the OData 2.0 syntax, except any statements with attribute <code>lastModifiedDateTime</code>. To learn more, see:</p> <ul style="list-style-type: none"> <li>• <a href="#">OData version 2</a> → <b>4.5. Filter System Query Option (\$filter)</b></li> <li>• <a href="#">SAP SuccessFactors HXM Suite OData API: Reference Guide (V2)</a> → <b>DynamicGroup</b></li> <li>• <a href="#">SAP SuccessFactors Workforce SCIM API and System for Cross-domain Identity Management for Workforce in SuccessFactors</a></li> </ul>
<code>sf.group.unique.attribute</code>	<p>If the service tries to create a group that already exists in the target system, the creation will fail. In this case, the existing group only needs to be updated. This group can be found via search, based on an attribute (default or specific).</p> <p>To make the search filter by a specific attribute, specify this attribute as a value for the <code>sf.group.unique.attribute</code> property.</p> <p>If the property is not specified, the search is done by the default attribute: <code>displayName</code></p> <p><b>Connector version:</b> SAP SuccessFactors version 2</p>
(Optional) <code>sf.page.size</code>	<p>Defines the paging size.</p> <ul style="list-style-type: none"> <li>• Default value: <b>100</b></li> <li>• Maximum value: <b>1000</b></li> </ul> <p><b>Connector version:</b> SAP SuccessFactors version 1</p>

Property Name	Description & Value
(Optional)sf.group.members.paging.enabled	<p>This property enables paging of group members.</p> <p>The maximum number of group members returned per request is 100. To read more than 100 group members, paging must be enabled.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><i>true</i> - Paging is enabled.</li> <li><i>false</i> - Paging is disabled.</li> </ul> <p>Default value: <i>false</i></p> <p><b>Connector version:</b> SAP SuccessFactors version 2</p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination (configuration):

```
Type=HTTP
Authentication=BasicAuthentication
ProxyType=Internet
URL=https://apitest.successfactors.com/odata/v2
User=sfsf_admin@mycompany.com
Password=*****

sf.user.attributes=userId,username,addressLine1,lastName,country,email,location,firstName,lastModifiedDateTime,personKeyNav,manager/username
sf.user.attributes.expand=personKeyNav,manager
sf.user.filter=department ne 'Manufacturing'
sf.group.filter=groupType eq 'permission'
sf.page.size=70
```

## 7. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [SAP SuccessFactors](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP SuccessFactors. For more information, see:

[Manage Transformations \[page 1494\]](#)

[SAP SuccessFactors HCM Suite OData API](#)



Default read and write transformations for SAP SuccessFactors HCM Suite OData API version 1:

→ Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.userId",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta .location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$ .username",
        "targetPath":
"$ .userName",
        "optional": true
      },
      {
        "sourcePath":
"$ .firstName",
        "targetPath":
"$ .name .givenName",
        "optional": true
      },
      {
        "sourcePath":
"$ .lastName",
        "targetPath":
"$ .name .familyName",
        "optional": true
      },
      {
        "sourcePath":
"$ .email",
        "targetPath":
"$ .emails[0].value",
        "optional": true
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:User",
        "targetPath":
"$ .schemas[0]"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "skipOperations": [
      "delete"
    ],
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.userId"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath": "$.userId",
        "scope": "createEntity"
      },
      {
        "constant": "t",
        "targetPath": "$ .status"
      },
      {
        "condition": "$ .active ==
false",
        "constant": "f",
        "targetPath": "$ .status"
      },
      {
        "sourcePath":
"$ .userName",
        "optional": true,
        "targetPath": "$ .username"
      },
      {
        "sourcePath":
"$ .name .familyName",
        "optional": true,
        "targetPath":
"$ .lastName",
        "defaultValue": null
      },
      {
        "sourcePath":
"$ .name .givenName",
        "optional": true,
        "targetPath":
"$ .firstName",
        "defaultValue": null
      },
      {
        "sourcePath":
"$ .name .honorificPrefix",
        "optional": true,
        "targetPath":
"$ .salutation",
        "defaultValue": null
      }
    ]
  }
}

```

```

    },
    {
      "sourcePath":
"$ .personKeyNav.personIdExternal",
      "targetPath":
"$ .externalId",
      "optional": true
    }
  ],
  {
    "group": {
      "scimEntityEndpoint":
"Groups",
      "mappings": [
        {
          "sourcePath": "$ .groupID",
          "targetVariable":
"entityIdSourceSystem",
          "targetPath": "$ .id"
        },
        {
          "sourcePath":
"$ .groupName",
          "targetPath":
"$ .displayName"
        },
        {
          "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
          "targetPath":
"$ .schemas[0]"
        },
        {
          "sourcePath":
"$ .users[*].userId",
          "preserveArrayWithSingleElement":
true,
          "targetPath": "$ .members[?
(@.value)]",
          "optional": true
        },
        {
          "constant": "User",
          "preserveArrayWithSingleElement":
true,
          "optional": true,
          "targetPath":
"$ .members[*].type"
        }
      ]
    }
  }
}

```

```

      "sourcePath":
"$ .name.honorificSuffix",
      "optional": true,
      "targetPath": "$ .suffix",
      "defaultValue": null
    },
    {
      "sourcePath":
"$ .emails[0].value",
      "optional": true,
      "targetPath": "$ .email",
      "defaultValue": null
    },
    {
      "condition": "$ .emails[?
(@.primary == true)].value != []",
      "sourcePath": "$ .emails[?
(@.primary == true)].value",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "defaultValue": null,
      "targetPath": "$ .email",
      "functions": [
        {
          "function":
"elementAt",
          "index": 0
        }
      ]
    },
    {
      "sourcePath":
"$ .timezone",
      "optional": true,
      "targetPath":
"$ .timeZone",
      "defaultValue": null
    },
    {
      "sourcePath":
"$ .nickName",
      "optional": true,
      "targetPath":
"$ .nickname",
      "defaultValue": null
    },
    {
      "sourcePath":
"$ .addresses[0].country",
      "optional": true,
      "targetPath": "$ .country",
      "defaultValue": null
    },
    {
      "condition":
"$ .addresses[?(@.primary ==
true)].country != []",
      "sourcePath":
"$ .addresses[?(@.primary ==
true)].country",

```

```

"preserveArrayWithSingleElement":
true,
    "optional": true,
    "defaultValue": null,
    "targetPath": "$.country",
    "functions": [
        {
            "function":
"elementAt",
            "index": 0
        }
    ]
},
{
    "sourcePath":
"$$.addresses[0].locality",
    "optional": true,
    "targetPath": "$.city",
    "defaultValue": null
},
{
    "condition":
"$$.addresses[?(@.primary ==
true)].locality != []",
    "sourcePath":
"$$.addresses[?(@.primary ==
true)].locality",
    "preserveArrayWithSingleElement":
true,
        "optional": true,
        "defaultValue": null,
        "targetPath": "$.city",
        "functions": [
            {
                "function":
"elementAt",
                "index": 0
            }
        ]
    },
    {
        "sourcePath":
"$$.addresses[0].formatted",
        "optional": true,
        "defaultValue": null,
        "targetPath":
"$$.addressLine1"
    },
    {
        "sourcePath":
"$$.addresses[1].formatted",
        "optional": true,
        "targetPath":
"$$.addressLine2",
        "defaultValue": null
    },
    {
        "sourcePath":
"$$.addresses[2].formatted",
        "optional": true,

```

```

        "targetPath":
        "$.addressLine3",
        "defaultValue": null
    },
    {
        "condition":
        "$.phoneNumbers[?(@.type ==
        'work')].value != []",
        "sourcePath":
        "$.phoneNumbers[?(@.type ==
        'work')].value",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "defaultValue": null,
        "targetPath":
        "$.businessPhone",
        "functions": [
            {
                "function":
                "elementAt",
                "index": 0
            }
        ]
    },
    {
        "condition":
        "$.phoneNumbers[?(@.type ==
        'fax')].value != []",
        "sourcePath":
        "$.phoneNumbers[?(@.type ==
        'fax')].value",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "defaultValue": null,
        "targetPath": "$.fax",
        "functions": [
            {
                "function":
                "elementAt",
                "index": 0
            }
        ]
    },
    {
        "condition":
        "$.phoneNumbers[?(@.type ==
        'mobile')].value != []",
        "sourcePath":
        "$.phoneNumbers[?(@.type ==
        'mobile')].value",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "defaultValue": null,
        "targetPath":
        "$.cellPhone",
        "functions": [

```

```

        {
            "function":
"elementAt",
            "index": 0
        }
    ],
    {
        "condition":
"$phoneNumbers[?(@.type ==
'home')].value != []",
        "sourcePath":
"$phoneNumbers[?(@.type ==
'home')].value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "defaultValue": null,
        "targetPath":
"$homePhone",
        "functions": [
            {
                "function":
"elementAt",
                "index": 0
            }
        ]
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true,
        "targetPath": "$empId",
        "defaultValue": null
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
        "optional": true,
        "targetPath":
"$division",
        "defaultValue": null
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']",
        "optional": true,
        "targetPath":
"$department",
        "defaultValue": null
    }
]
},
"group": {

```

```
    "scimEntityEndpoint":
    "Groups",
    "skipOperations": [
      "create",
      "delete"
    ],
    "mappings": [
      {
        "sourceVariable":
        "entityIdTargetSystem",
        "targetPath": "$.groupID"
      },
      {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.groupName"
      },
      {
        "constant":
        "DynamicGroup",
        "targetPath":
        "$.__metadata.uri"
      },
      {
        "constant": "permission",
        "targetPath":
        "$.groupType"
      },
      {
        "optional": true,

        "preserveArrayWithSingleElement":
        true,
        "sourcePath":
        "$.members[*].value",
        "targetPath": "$.members[?
        (@.value)]"
      }
    ]
  }
}
```

## Default read and write transformations for SCIM API version 2:

### Read Transformation

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetPath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$$.meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ]
      }
    ],
    "sourcePath":
"$$.schemas",
    "preserveArrayWithSingleElement":
true,
    "targetPath":
"$$.schemas"
  },
  {
    "sourcePath":
"$$.userName",
    "targetPath":
"$$.userName",
    "correlationAttribute": true
  },
  {
    "sourcePath":
"$$.userType",
    "targetPath":
"$$.userType"
  },
  {
    "sourcePath":
"$$.name.familyName",
    "optional": true,
    "targetPath":
"$$.name.familyName"
  }
]
```

### Write Transformation

#### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "constant": [
"urn:ietf:params:scim:schemas:extension:successfactors:2.0:User",
"urn:ietf:params:scim:schemas:core:2.0:User",
"urn:ietf:params:scim:schemas:extension:sap:2.0:User",
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
],
        "targetPath":
"$$.schemas"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName"
      },
      {
        "sourcePath":
"$$.nickName",
        "optional": true,
        "targetPath":
"$$.nickName"
      },
      {
        "sourcePath":
"$$.preferredLanguage",
        "optional": true,
        "targetPath":
"$$.preferredLanguage"
      },
      {
        "sourcePath":
"$$.userType",
        "targetPath":
"$$.userType"
      },
      {
        "sourcePath":
"$$.externalId",
        "optional": true,

```



```

    {
      "sourcePath":
"$$.name.givenName",
      "optional": true,
      "targetPath":
"$$.name.givenName"
    },
    {
      "sourcePath":
"$$.name.middleName",
      "optional": true,
      "targetPath":
"$$.name.middleName"
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "optional": true,
      "targetPath":
"$$.name.honorificPrefix"
    },
    {
      "sourcePath":
"$$.name.honorificSuffix",
      "optional": true,
      "targetPath":
"$$.name.honorificSuffix"
    },
    {
      "sourcePath":
"$$.name.formatted",
      "optional": true,
      "targetPath":
"$$.name.formatted"
    },
    {
      "sourcePath":
"$$.nickName",
      "optional": true,
      "targetPath":
"$$.nickName"
    },
    {
      "sourcePath":
"$$.preferredLanguage",
      "optional": true,
      "targetPath":
"$$.preferredLanguage"
    },
    {
      "sourcePath":
"$$.displayName",
      "optional": true,
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath":
"$$.title",
      "optional": true,
      "targetPath":
"$$.title"

```

```

      "targetPath":
"$$.externalId"
    },
    {
      "sourcePath":
"$$.displayName",
      "optional": true,
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath":
"$$.locale",
      "optional": true,
      "targetPath":
"$$.locale"
    },
    {
      "sourcePath": "$[
'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
      "optional": true,
      "targetPath": "$[
'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']['userUuid']"
    },
    {
      "sourcePath":
"$$.emails",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$$.emails"
    },
    {
      "condition":
"$$.emails.length() > 0",
      "targetPath":
"$$.emails[*].type",
      "constant": "work"
    },
    {
      "sourcePath":
"$$.timezone",
      "optional": true,
      "targetPath":
"$$.timezone"
    },
    {
      "sourcePath":
"$$.name.formatted",
      "optional": true,
      "targetPath":
"$$.name.formatted"
    },
    {
      "sourcePath":
"$$.name.familyName",
      "optional": true,

```

```

    },
    {
      "sourcePath":
"$$.externalId",
      "optional": true,
      "targetPath":
"$$.externalId"
    },
    {
      "sourcePath":
"$$.locale",
      "optional": true,
      "targetPath":
"$$.locale"
    },
    {
      "sourcePath":
"$$.timezone",
      "optional": true,
      "targetPath":
"$$.timezone"
    },
    {
      "sourcePath": "$[
'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
      "optional": true,
      "targetPath": "$[
'urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']['userUuid']"
    },
    {
      "sourcePath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['perPersonUuid']",
      "optional": true,
      "targetPath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['perPersonUuid']"
    },
    {
      "sourcePath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['loginMethod']",
      "optional": true,
      "targetPath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['loginMethod']"
    },
    {
      "sourcePath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['personIdExternal']",
      "optional": true,
      "targetPath": "$[
'urn:ietf:params:scim:schemas:ext

```

```

      "targetPath":
"$$.name.familyName"
    },
    {
      "sourcePath":
"$$.name.givenName",
      "optional": true,
      "targetPath":
"$$.name.givenName"
    },
    {
      "sourcePath":
"$$.name.middleName",
      "optional": true,
      "targetPath":
"$$.name.middleName"
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "optional": true,
      "targetPath":
"$$.name.honorificPrefix"
    },
    {
      "sourcePath":
"$$.name.honorificSuffix",
      "optional": true,
      "targetPath":
"$$.name.honorificSuffix"
    },
    {
      "sourcePath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['perPersonUuid']",
      "optional": true,
      "targetPath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['perPersonUuid']"
    },
    {
      "sourcePath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['loginMethod']",
      "optional": true,
      "targetPath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['loginMethod']"
    },
    {
      "sourcePath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['personIdExternal']",
      "optional": true,
      "targetPath": "$[
'urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['personIdExternal']"
    }
  ]
}

```

```

extension:successfactors:2.0:User']
['personIdExternal']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['customFields']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['customFields']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext

```

```

    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['customFields']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:successfactors:2.0:User']
['customFields']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['department']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['value']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
      "optional": true,
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['$ref']",
      "optional": true,

```

```

extension:enterprise:2.0:User']
['manager']['$ref']",
    "optional": true,
    "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['$ref']"
    },
    {
        "sourcePath":
"$$.phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.phoneNumbers"
    },
    {
        "sourcePath":
"$$.emails",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.emails"
    },
    {
        "sourcePath":
"$$.emails[?(@.primary==
true)].value",
        "optional": true,
        "correlationAttribute": true
    },
    {
        "sourcePath":
"$$.active",
        "targetPath":
"$$.active"
    },
    {
        "sourcePath":
"$$.groups",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.groups"
    },
    {
        "scimEntityEndpoint":
"Users"
    },
    {
        "group": {
            "mappings": [
                {
                    "sourcePath":
"$$.id",

```

```

        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['$ref']"
    },
    {
        "sourcePath":
"$$.title",
        "optional": true,
        "targetPath":
"$$.title"
    },
    {
        "sourcePath":
"$$.phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.phoneNumbers"
    },
    {
        "sourcePath":
"$$.active",
        "targetPath":
"$$.active"
    },
    {
        "sourcePath":
"$$.groups[*].value",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.groups[?(@.value)]",
        "functions": [
            {
                "entityType": "group",
                "function": "resolveEntityIds"
            }
        ],
        "sourcePath":
"$$.Operations",
        "preserveArrayWithSingleElement":
true,
        "targetPath":
"$$.Operations",
        "scope":
"patchEntity"
    },
    {
        "sourcePath":
"$$.schemas",
        "preserveArrayWithSingleElement":
true,

```

```

        "targetPath":
"$ .id",
        "targetVariable":
"entityIdSourceSystem",
        "correlationAttribute": true
    },
    "sourceVariable":
"entityBaseLocation",
    "targetPath":
"$ .meta.location",
    "targetVariable":
"entityLocationSourceSystem"
    },
    {
        "sourcePath":
"$ .schemas",
        "targetPath":
"$ .schemas"
    },
    {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName"
    },
    {
        "sourcePath":
"$ .members",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$ .members"
    }
    ],
    "scimEntityEndpoint":
"Groups"
}

```

```

        "targetPath":
"$ .schemas",
        "scope":
"patchEntity"
    },
    "scimEntityEndpoint":
"Users"
    },
    "group": {
        "skipOperations": [
            "create",
            "delete"
        ],
        "mappings": [
            {
                "sourceVariable":
"entityIdTargetSystem",
                "targetPath":
"$ .id"
            },
            {
                "sourcePath":
"$ .Operations",
                "preserveArrayWithSingleElement":
true,
                "targetPath":
"$ .Operations",
                "scope":
"patchEntity"
            },
            {
                "sourcePath":
"$ .displayName",
                "targetPath":
"$ .displayName"
            },
            {
                "sourcePath":
"$ .schemas",
                "preserveArrayWithSingleElement":
true,
                "targetPath":
"$ .schemas",
                "scope":
"patchEntity"
            },
            {
                "constant": [
                    "urn:ietf:params:scim:schemas:core:2.0:Group",
                    "urn:sap:cloud:scim:schemas:extension:custom:2.0:Group"
                ],
                "targetPath":
"$ .schemas"
            }
        ]
    }
}

```

```

    "targetPath":
    "$.members",
    "type": "remove"
  },
  {
    "sourcePath":
    "$.members[*].value",
    "preserveArrayWithSingleElement":
    true,
    "optional": true,
    "targetPath":
    "$.members[?(@.value)]"
  }
],
"scimEntityEndpoint":
"Groups"
}

```

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[URI Conventions \(OData Version 2.0\)](#) 

[SAP SuccessFactors HCM Suite OData API](#)

[SAP SuccessFactors Workforce SCIM API](#)

## 1.6.3.33 SAP SuccessFactors Learning

Follow this procedure to set up SAP SuccessFactors Learning as a proxy system.

### Prerequisites

- You have OAuth credentials for SAP SuccessFactors Learning. For more information, see: [Generating Client ID & Secret](#)

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

SAP SuccessFactors Learning is a learning solution which helps organizations to improve employee skills and talent management, align learning outcomes with performance goals, boost compliance, and train external audiences.

You can use the Identity Provisioning user interface (UI) to connect to SAP SuccessFactors Learning as a proxy system and configure it in hybrid scenarios. For example, when SAP SuccessFactors Learning is exposed as a proxy system, you can connect it to an external identity management system, such as SAP Identity Management, without making a direct connection between both systems. You can provision users and groups to the external backend system, which can trigger CRUD (create, read, update, delete) operations on users and group members back to the SAP SuccessFactors Learning.

#### ! Restriction

Provisioning **groups** is not supported.

### Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit ► [Neo](#) ► [Overview](#) , you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add *SAP SuccessFactors Learning* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.




## Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the API of your SAP SuccessFactors Learning system. It follows the pattern: <code>https://&lt;root URL&gt;/learning/public-api/rest/admin/Integration.svc/ias</code>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the technical user ID for SAP SuccessFactors Learning.
Password	(Credential) Enter the password for the SAP SuccessFactors Learning technical user. For more information, see <a href="#">Learning Technical User</a> .
(Optional) <code>lms.user.filter</code>	<p>When specified, only those users matching the filter expression will be read.</p> <p><b>Possible values:</b></p> <ul style="list-style-type: none"> <li><code>userName eq "testName"</code></li> <li><code>externalID eq "testID"</code></li> <li><code>active eq "true"</code></li> <li><code>sourceSystem eq "Learning"</code> - indicates that the user is created directly in SAP SuccessFactors Learning with no involvement of Identity Provisioning.</li> <li><code>sourceSystem eq "Identity Provisioning"</code> - indicates that the user is created in SAP SuccessFactors Learning by Identity Provisioning.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

## SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.

- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '**tooMany**'.

Bear in mind the following restrictions:

- In the **Read Transformation**, there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
*GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.emails[0].value*, *\$.emails[1].value*, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (*\$.emails[0].value*).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the **Properties** tab: *scim.user.filter = timezone eq "Africa"*

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SAP SuccessFactors Learning API will result into: **/Users?filter=timezone eq "Africa" and userName eq "johnsmith03"**

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the **SAP SuccessFactors Learning** proxy system, whose settings are displayed under the **Transformations** tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SAP SuccessFactors Learning system. For more information, see:

[Manage Transformations \[page 1494\]](#).

SCIM API URL

Default read and write transformations:

#### → Tip

The proxy **Read Transformation** is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application

can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

        "sourcePath":
"$ .userName",
        "optional": true,
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
    },
    {
        "sourcePath":
"$ .externalId",
        "optional": true,
        "targetPath":
"$ .externalId"
    },
    {
        "sourcePath": "$ .locale", {
"user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
        {
            "sourcePath": "$ .id",
            "targetPath": "$ .id",
            "targetVariable":
"entityIdSourceSystem"
        },
        {
            "sourceVariable":
"entityBaseLocation",
            "targetPath":
"$ .meta.location",
            "targetVariable":
"entityLocationSourceSystem",
            "functions": [
                {
                    "optional": true,
                    "targetPath": "$ .locale"
                }
            ],
            {
                "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
                "optional": true,
                "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['userUuid']"
            },
            {
                "sourcePath": "$
['urn:sap:cloud:scim:schemas:exten
sion:custom:2.0:User'] ['siteID']",
                "optional": true,
                "targetPath": "$
['urn:sap:cloud:scim:schemas:exten
sion:custom:2.0:User'] ['siteID']"
            },
            {

```

## Code Syntax

```

{
    "user": {
        "scimEntityEndpoint": "Users",
        "mappings": [
            {
                "sourceVariable":
"entityIdTargetSystem",
                "targetPath": "$ .id"
            },
            {
                "sourcePath": "$ .schemas",
                "preserveArrayWithSingleElement":
true,
                "targetPath": "$ .schemas"
            },
            {
                "sourcePath":
"$ .userName",
                "optional": true,
                "targetPath": "$ .userName"
            },
            {
                "sourcePath":
"$ .externalId",
                "optional": true,
                "targetPath":
"$ .externalId"
            },
            {
                "sourcePath": "$ .locale",
                "optional": true,
                "targetPath": "$ .locale"
            },
            {
                "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUuid']",
                "optional": true,
                "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['userUuid']"
            },
            {
                "sourcePath": "$
['urn:sap:cloud:scim:schemas:exten
sion:custom:2.0:User'] ['siteID']",
                "optional": true,
                "targetPath": "$
['urn:sap:cloud:scim:schemas:exten
sion:custom:2.0:User'] ['siteID']"
            },
            {
                "sourcePath": "$
['urn:sap:cloud:scim:schemas:exten
sion:custom:2.0:User']
['customColumns'] ['110']",
                "optional": true,

```

```

      "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ]
[ 'sourceSystem' ]",
      "optional": true,
      "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ]
[ 'sourceSystem' ]"
    },
    {
      "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ]
[ 'applicationID' ]",
      {
        "type":
"concatString",
        "suffix": "$
{entityIdSourceSystem}"
      }
    },
    {
      "sourcePath": "$.schemas",

"preserveArrayWithSingleElement":
true,
      "targetPath": "$.schemas"
    },
    {
      "sourcePath":
"$ .userName",
      "optional": true,
      "targetPath":
"$ .userName",
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$ .externalId",
      "optional": true,
      "targetPath":
"$ .externalId"
    },
    {
      "sourcePath": "$.locale",
      "optional": true,
      "targetPath": "$.locale"
    },
    {
      "sourcePath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
[ 'userUid' ]",
      "optional": true,
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:extension:sap:2.0:User' ]
[ 'userUid' ]"
    },
    {

```

```

      "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ]
[ 'customColumns' ][ '110' ]"
    },
    {
      "sourcePath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ]
[ 'customColumns' ][ '120' ]",
      "optional": true,
      "targetPath": "$
[ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:User' ]
[ 'customColumns' ][ '120' ]"
    },
    {
      "condition": "$.emails[?
(@.primary== true)] empty false",
      "sourcePath": "$.emails[?
(@.primary== true)].value",

"preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath": "$.emails[?
(@.value)]"
    },
    {
      "condition": "$.emails[?
(@.primary== true)] empty true",
      "sourcePath":
"$ .emails[0].value",
      "targetPath":
"$ .emails[0].value",

"preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$ .name.givenName",
      "optional": true,
      "targetPath":
"$ .name.givenName"
    },
    {
      "sourcePath":
"$ .name.middleName",
      "optional": true,
      "targetPath":
"$ .name.middleName"
    },
    {
      "sourcePath":
"$ .name.familyName",
      "targetPath":
"$ .name.familyName"
    },
    {

```

```

      "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['siteID'],
      "optional": true,
      "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['siteID']
      "optional": true,
      "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['applicationID']"
    },
    {
      "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['110'],
      "optional": true,
      "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['110']"
    },
    {
      "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['120'],
      "optional": true,
      "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['120']"
    },
    {
      "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['sourceSystem'],
      "optional": true,
      "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['sourceSystem']"
    },
    {
      "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['applicationID'],
      "optional": true,
      "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['applicationID']"
    },
    {
      "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['110'],

```

```

      "condition":
"$phoneNumbers[?(@.primary==true)] empty false",
      "sourcePath":
"$phoneNumbers[?(@.primary==true)].value",

      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath":
"$phoneNumbers[?(@.value)]"
    },
    {
      "condition":
"$phoneNumbers[?(@.primary==true)] empty true",
      "sourcePath":
"$phoneNumbers[0].value",
      "targetPath":
"$phoneNumbers[0].value",

      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath": "$.active",
      "targetPath": "$.active"
    },
    {
      "sourcePath":
$.Operations,
      "targetPath":
$.Operations,

      "preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    },
    {
      "sourcePath": $.schemas,
      "targetPath": $.schemas,

      "preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    }
  ]
}

```

```

        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['110']"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['120']]"
    },
    {
        "sourcePath": "$.emails",

"preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$.emails"
    },
    {
        "sourcePath": "$.emails[?
(@.primary== true)].value",
        "correlationAttribute":
true
    },
    {
        "sourcePath":
"$ .name.givenName",
        "optional": true,
        "targetPath":
"$ .name.givenName"
    },
    {
        "sourcePath":
"$ .name.middleName",
        "optional": true,
        "targetPath":
"$ .name.middleName"
    },
    {
        "sourcePath":
"$ .name.familyName",
        "targetPath":
"$ .name.familyName",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']
['customColumns']['120']"
    },
    {
        "sourcePath": "$.emails",
    },
    {
        "sourcePath":
"$ .phoneNumbers",

"preserveArrayWithSingleElement":
true,
        "optional": true,

```

```

        "targetPath":
"$ .phoneNumbers"

"preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath": "$.emails"
    },
    {
        "sourcePath": "$.emails[?
(@.primary== true)].value",
        "correlationAttribute":
true
    },
    {
        "sourcePath":
"$ .name.givenName",
        "optional": true,
        "targetPath":
"$ .name.givenName"
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active"
    }
    {
        "sourcePath":
"$ .name.middleName",
        "optional": true,
        "targetPath":
"$ .name.middleName"
    },
    {
        "sourcePath":
"$ .name.familyName",
        "targetPath":
"$ .name.familyName"
    },
    {
        "sourcePath":
"$ .phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$ .phoneNumbers"
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active"
    }
    ]
}

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.



If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## 1.6.3.34 Sales Cloud – Analytics & AI

Follow this procedure to set up Sales Cloud – Analytics & AI as a proxy system.

### Prerequisites

- You have technical user credentials for an Sales Cloud Analytics & AI (in short, [SCAAI](#)) system with read and write access permissions.


#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

Create a SCIM 2.0 proxy connector for Sales Cloud – Analytics & AI to execute hybrid scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to the SCAAI system, whenever the external back end requests such. This scenario supports provisioning **users** and **user assignments to groups**.

#### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use '**eq**' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used '**eq**' filter).
- Fully qualified names ([<schema>:<attribute>](#)) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)

- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `sales.cloud.analytics_ai.user.filter = ExternalId eq "Smith_03"`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=userName eq "johnsmith03"`

The query request to the SCAA API will result into: `/Users?filter=ExternalId eq "Smith_03" and userName eq "johnsmith03"`

## Procedure

1. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

#### SAP Cloud Identity Infrastructure

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → [SAP Cloud Identity Infrastructure](#)

#### Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → [SAP BTP, Neo Environment](#)

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [Sales Cloud – Analytics & AI](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the SCIM API portal of your SCAAI system.

Property Name	Value
ProxyType	Enter: <i>Internet</i>
Authentication	Enter: <i>BasicAuthentication</i>
User	Enter the user for your SCAAI system.
Password	Enter the password for your SCAAI user.
OAuth2TokenServiceURL	Enter the URL to the OAuth2 token service.  If not sure about the exact URL, ask your SCAAI administrator.
(Op-tio-nal)sales.cloud.analytics_ai.group.filter	Enter a group filter criteria, according to the API syntax of SCAAI.  For example: <b>displayName eq "first_group"</b>
(Op-tio-nal)sales.cloud.analytics_ai.user.filter	Enter a user filter criteria, according to the API syntax of SCAAI.  For example: <b>externalId eq "John123"</b>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

Exemplary destination:

Type=*HTTP*

Authentication=*BasicAuthentication*

ProxyType=*Internet*

URL=*http://myscaai:8080/scim\_services*

User=*MySCAAIUser*

Password=*\*\*\*\*\**

OAuth2TokenServiceURL=*http://myscaai:8080/gateway\_services/api/auth/ips/token*

## 6. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Sales Cloud – Analytics & AI* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your Sales Cloud – Analytics & AI. For more information, see [Manage Transformations \[page 1494\]](#).

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath": "$.schemas",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",

"preserveArrayWithSingleElement":
true,
        "optional" : true
      },
      {
        "type": "remove",
        "targetPath":
"$ .groups[*].display"
      },
      {
        "type": "remove",
        "targetPath":
"$ .groups[*].ref"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$ .id"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .externalId"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:User",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName"
      }
    ],
    "scimEntityEndpoint":
"Users"
  },
  "group": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$ .id"
      },
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .externalId"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName"
      }
    ]
  }
}

```

```

    ]
  },
  "group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta .location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName"
      },
      {
        "optional": true,

"preserveArrayWithSingleElement":
true,
        "sourcePath": "$.members",
        "targetPath": "$.members"
      },
      {
        "type": "remove",
        "targetPath":
"$ .members[*].$ref"
      },
      {
        "type": "remove",
        "targetPath":
"$ .members[*].display"
      }
    ]
  }
}

```

```

    "sourcePath":
"$ .members[*].value",

"preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetPath":
"$ .members[?(@.value)]"
  },
  "scimEntityEndpoint":
"Groups"
}

```



```
}
```

If you want the users and groups in SCAAI to have the **same** IDs as the respective users and groups in the external back-end system, modify the [Write Transformation](#) mappings as follows:

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.externalId"
      },
      ...
    ]
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.externalId"
      },
      ...
    ]
  },
  ...
}
```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

#### Neo Environment

- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>• For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul> | <ul style="list-style-type: none"> <li>• For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>• For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PATCH</b>.</li> </ul> |
|---|--|

#### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PATCH** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is

configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

#### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

### 1.6.3.35 Cloud Foundry UAA Server

Follow this procedure to set up the Cloud Foundry UAA server as a proxy system.

## Prerequisites

#### ! Restriction

This system is available for standalone tenants running on SAP Cloud Identity infrastructure and SAP BTP, Neo environment. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it only through **SAP Jam Collaboration** and **SAP Identity Access Governance** bundle options.

- You have a technical user with administrator permissions for Cloud Foundry UAA to read, create, and update user account information. You need Cloud Foundry UAA version **4.2** or higher.
- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need to do this only if the Cloud Foundry UAA server is exposed in a private corporate network. For more information, see [Cloud Connector](#).

## i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

User Account and Authentication Service (UAA) is an OAuth2 server that you can use for centralized identity management. It owns the user accounts and authentication sources, and supports standard protocols (such as [SAML](#), [LDAP](#), and [OpenID Connect](#)) to provide SSO and delegated authorization to Web applications. For more information, see [Cloud Foundry UAA: Overview](#) .

Cloud Foundry UAA is responsible for the SAP ID service to create and manage platform users (platform administrators and platform developers) in Cloud Foundry.

### → Tip

This connector is meant for provisioning users and groups from/to **general** Cloud Foundry systems (they could be non-SAP ones). If you want to trigger provisioning of entities from/to SAP Business Technology Platform Cloud Foundry applications, you'd better use [SAP BTP XS Advanced UAA \(Cloud Foundry\) \[page 1129\]](#) proxy system.

These proxy systems consume SCIM 1.1 API provided by Cloud Foundry UAA.

### → Remember

You can manage users and groups to Cloud Foundry on **application** level only. You cannot manage them on a [subaccount](#) level.

## SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#) standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).

- Fully qualified names (**<schema>:<attribute>**) are not supported. For example:  
`GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber eq '<attribute>'`
- If your system supports multivalued e-mails (that is `$.emails[0].value`, `$.emails[1].value`, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (`$.emails[0].value`).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the *Properties* tab: `scim.user.filter = timezone eq "US"`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=userName eq "johnsmith03"`

The query request to the Cloud Foundry UAA API will result into: `/Users?filter=timezone eq "US" and userName eq "johnsmith03"`

Follow the steps below to create a SCIM 2.0 representation of your proxy Cloud Foundry UAA system. You can then provision on demand new users and groups back to Cloud Foundry UAA.

## Procedure

- (Optional) Open Cloud Connector to add an access control system mapping for the **Cloud Foundry UAA Server**. This is needed to allow the Identity Provisioning service to access the Cloud Foundry UAA system as a back-end system on the intranet. To learn how, see: [Configure Access Control \(HTTP\)](#)
- Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → *Neo* → *Overview*, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

- Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

4. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
5. Add [Cloud Foundry UAA Server](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
6. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Specify the URL to the Cloud Foundry UAA SCIM API.  If not sure about the exact URL, ask your Cloud Foundry UAA administrator.
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
OAuth2TokenServiceURL	As you need to make OAuth authentication to the UAA system, enter the URL to the OAuth2 token service.  If not sure about the exact URL, ask your Cloud Foundry UAA administrator.
User	Enter the OAuth client ID of the Cloud Foundry UAA technical user.
Password	(Credential) Enter the OAuth client secret of the technical user.
uaa.origin	Enter the location of your Cloud Foundry identity provider. If not sure about the value, ask your Cloud Foundry UAA administrator.  The value of this property is a string, which will be used as the <a href="#">origin</a> attribute in the system transformations.
uaa.origin.filter.enabled	This flag property depends on <code>uaa.origin</code> . Possible values: <b>true</b> or <b>false</b> <ul style="list-style-type: none"> <li>If set to <a href="#">true</a>, the Identity Provisioning service will read only users whose identity provider is set as a value of <code>uaa.origin</code>.</li> <li>If set to <a href="#">false</a>, the Identity Provisioning service will read all users, regardless of their origin.</li> <li>If set to <a href="#">true</a> but the <code>uaa.origin</code> property is missing, the provisioning will fail.</li> </ul>
scim.support.patch.operation	Use this property if you want to modify the members of a group.  Possible values: <ul style="list-style-type: none"> <li><b>true</b> – the Identity Provisioning service will modify the group membership via the <a href="#">PATCH /Groups</a> endpoint of UAA. To learn how, see <a href="#">Patch</a> 📖</li> <li><b>false</b> – the Identity Provisioning service will modify the group membership via the <a href="#">POST /Groups</a> or <a href="#">DELETE /Groups</a> endpoints of UAA. To learn how, see <a href="#">Add Member</a> 📖 and <a href="#">Remove Member</a> 📖.</li> </ul>

Property Name	Description & Value
<code>uaa.patch.response.with.resource</code>	<p>Use this property if you want to retrieve a group whose membership was modified.</p> <div> <p><b>Note</b></p> <p>This property is usable only when you have configured membership modifications via <a href="#">Add/Remove Member</a> UAA endpoints. That is, when the <code>scim.support.patch.operation</code> property is set to <b>false</b>.</p> </div> <p>Possible values:</p> <ul style="list-style-type: none"> <li><b>true</b> – the Identity Provisioning service will return the modified group via the <a href="#">GET /Groups</a> endpoint of UAA. To learn how, see <a href="#">Retrieve</a>.</li> <li><b>false</b> – no modified groups will be returned by the service.</li> </ul>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

Exemplary destination:

Type=[HTTP](#)

Authentication=[BasicAuthentication](#)

ProxyType=[Internet](#)

URL=<https://api.authentication.hana.ondemand.com>

OAuth2TokenServiceURL=<https://MyCFaccount.authentication.hana.ondemand.com/oauth/token>

User=[MyCFuser](#)

Password=\*\*\*\*\*

`uaa.origin`=[my\\_UAA\\_location](#)

`uaa.origin.filter.enabled`=[true](#)

`scim.support.patch.operation`=[true](#)

`uaa.patch.response.with.resource`=[false](#)

## 7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [Cloud Foundry UAA Server](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your Cloud Foundry UAA server. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Cloud Foundry UAA API: Users](#) ➡

[Cloud Foundry UAA API: Groups](#) ➡

Default read and write transformations:

#### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.



## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
"entityIdSourceSystem"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetPath":
"$$.meta.location",
        "targetVariable":
"entityLocationSourceSystem",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath":
"$$.userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement":
true
      },
      {
        "sourcePath": "$.emails[?
(@.primary== true)].value",
        "correlationAttribute":
true
      },
      {
        "sourcePath": "$.groups",
        "targetPath": "$.groups",
        "preserveArrayWithSingleElement":
true,
        "optional": true
      },
    ]
  },

```

## Code Syntax

```

{
  "user": {
    "mappings": [
      {
        "constant": "uaa-dummy-
value",
        "targetPath": "$.id",
        "scope": "createEntity"
      },
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "sourcePath":
"$$.userName",
        "targetPath": "$.userName"
      },
      {
        "sourcePath": "$.name",
        "targetPath": "$.name",
        "optional": true
      },
      {
        "sourcePath": "$.emails",
        "targetPath": "$.emails",
        "preserveArrayWithSingleElement":
true
      },
      {
        "condition": "$.emails[?
(@.primary == true)].value == []",
        "targetPath":
"$$.emails[0].primary",
        "constant": true
      },
      {
        "sourcePath":
"$$.phoneNumbers",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath":
"$$.phoneNumbers"
      },
      {
        "sourcePath":
"$$.externalId",
        "optional": true,
        "targetPath":
"$$.externalId"
      },
      {
        "sourcePath":
"$$.verified",
        "targetPath":
"$$.verified",

```

```

    {
      "sourcePath":
"$ .phoneNumbers",
      "targetPath":
"$ .phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath": "$ .active",
      "targetPath": "$ .active",
      "optional": true
    },
    {
      "sourcePath": "$ .meta",
      "targetPath": "$ .meta",
      "optional": true
    },
    {
      "sourcePath":
"$ .externalId",
      "targetPath":
"$ .externalId",
      "optional": true
    },
    {
      "sourcePath": "$ .origin",
      "targetPath": "$ .origin",
      "optional": true
    },
    {
      "sourcePath": "$ .zoneId",
      "targetPath": "$ .zoneId",
      "optional": true
    },
    {
      "sourcePath":
"$ .verified",
      "targetPath":
"$ .verified",
      "optional": true
    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
      "targetPath":
"$ .schemas[0]"
    },
    ],
    "scimEntityEndpoint": "Users"
  },
  "group": {
    "mappings": [
      {
        "sourcePath": "$ .id",
        "targetPath": "$ .id",
        "targetVariable":
"entityIdSourceSystem"
      },

```

```

      "optional": true
    },
    {
      "constant":
"%uaa.origin%",
      "targetPath": "$ .origin"
    },
    {
      "constant":
"urn:scim:schemas:core:1.0",
      "targetPath":
"$ .schemas[0]"
    },
    ],
    "scimEntityEndpoint": "Users"
  },
  "group": {
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$ .id"
      },
      {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .displayName"
      },
      {
        "sourcePath":
"$ .description",
        "targetPath":
"$ .description",
        "optional": true
      },
      {
        "constant":
"urn:scim:schemas:core:1.0",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "sourcePath": "$ .members",
        "targetPath": "$ .members",
      },
    ],
    "preserveArrayWithSingleElement":
true,
    "optional": true,
    "functions": [
      {
        "condition": "@.type
EMPTY false",
        "function":
"toUpperCaseString",
        "applyOnElements":
true,
        "applyOnAttribute":
"type",
        "locale": "en_EN"
      },
    ],
  },

```

```

    {
      "sourceVariable":
"entityBaseLocation",
      "targetPath":
"$$.meta.location",
      "targetVariable":
"entityLocationSourceSystem",
      "functions": [
        {
          "type":
"concatString",
          "suffix": "$
{entityIdSourceSystem}"
        }
      ],
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath":
"$$.description",
      "targetPath":
"$$.description",
      "optional": true
    },
    {
      "sourcePath": "$.members",
      "targetPath": "$.members",

"preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath": "$.zoneId",
      "targetPath": "$.zoneId"
    },
    {
      "sourcePath": "$.meta",
      "targetPath": "$.meta",
      "optional": true
    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
      "targetPath":
"$$.schemas[0]"
    }
  ],
  "scimEntityEndpoint": "Groups"
}

```

```

    {
      "constant":
"%uaa.origin%",
      "targetPath":
"$$.members[*].origin",

"preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.Operations",
      "targetPath":
"$$.Operations",

"preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    },
    {
      "sourcePath": "$.schemas",
      "targetPath": "$.schemas",

"preserveArrayWithSingleElement":
true,
      "scope": "patchEntity"
    }
  ],
  "scimEntityEndpoint": "Groups"
}

```

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Cloud Foundry UAA: Users](#) ➦

[Cloud Foundry UAA: Groups](#) ➦

### 1.6.3.36 Google G Suite

Follow this procedure to set up Google G Suite as a proxy system.

#### Prerequisites

1. Sign in to the Google API console (<https://console.developers.google.com> ➦) and create a project.
2. Enable the Admin SDK. To do this, go to ► [Dashboard](#) ► [ENABLE API](#) ► [Admin SDK](#) ► [ENABLE](#) ►.
3. Create a service account for your project. We recommend that you select [Enable G Suite Domain-wide Delegation](#) during the creation. If you skip this option, you can set it later. For more information, see [Creating a service account](#) ➦.
4. Then, in the Google admin console (<https://admin.google.com> ➦), a user with **Super Admin** role can delegate domain-wide authority to your service account. This way, it will have access to the Google Admin SDK on behalf of your user. For more information, see [Delegating domain-wide authority](#) ➦.

**NOTE:** When specifying the scopes, the administrator has to enter the following:

<https://www.googleapis.com/auth/admin.directory.user>, <https://www.googleapis.com/auth/admin.directory.group>

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context

A Google service account with delegated domain-wide authority is required for authentication and authorization of the Identity Provisioning service to G Suite domain. The authentication is based on OAuth 2.0 protocol with JSON Web Token (JWT). The private key for the signature is distributed by Google via one-time downloadable JSON data, which is accessible by the domain administrator. The private key is encoded in PKCS8 format and is in the [private\\_key](#) field of the JSON data. For more information, see [JSON Web Token \(JWT\)](#) ➦.

- When using it as a source system, you can read both users and groups from Google G Suite and provision them to any target system you have added in the Identity Provisioning user interface.
- When using it as a target system, you can write both users and groups, read from any source system you have added in the Identity Provisioning user interface. Google G Suite can automatically create accounts for your users in the Google Cloud Datastore.

The Identity Provisioning service supports user and group operations based on the following Google Directory API. See the table below.

User Operations	Group Operations
<a href="#">Create a user</a> ➡	<a href="#">Create a group</a> ➡
<a href="#">Retrieve a user</a> ➡	<a href="#">Retrieve a group's properties</a> ➡
<a href="#">Update a user</a> ➡	<a href="#">Update a group's properties</a> ➡
<a href="#">Delete a user</a> ➡	<a href="#">Delete a group</a> ➡

### ⚠ Caution

You can only provision users whose e-mails are from verified domains.

If you have successfully finished with the initial setup (described in the **Prerequisites** section), continue with the procedure below.

## Procedure

- Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

- Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [Google G Suite](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>

Property Name	Description & Value
URL	Specify the service URL:  <a href="https://www.googleapis.com/admin/directory">https://www.googleapis.com/admin/directory</a>
ProxyType	Enter: <b>Internet</b>
Authentication	Enter: <i>BasicAuthentication</i>  The authentication type in use is actually <b>OAuth</b> with JWT. But for any provisioning system based on OAuth, <b>BasicAuthentication</b> is used along with the OAuth2TokenServiceURL additional property.
User	Enter the service account's ID. You can take it from the " <i>client_email</i> " field in the JSON data, downloaded during the setup of Google service account.
Password	Enter the service account's private key, which represents a long string in PKCS8 format. You can take it from the " <i>private key</i> " field in the JSON data, downloaded during the setup of Google service account.
OAuth2TokenServiceURL	To make OAuth authentication to the Google G Suite system, enter the URL to the access token provider service. For more information, see Using <a href="#">OAuth 2.0 to Access Google APIs</a> .
jwt.subject	Enter the Google G Suite user on behalf of which the Google Directory API is called. This user has been assigned the role <b>User Management Admin</b> .  This property corresponds to "sub" claim in JWT being generated during access token request: JWT: "sub" (Subject) Claim .
(Optional) jwt.scope	Enter space-separated Google Directory API authorization scopes. For example:  <a href="https://www.googleapis.com/auth/admin.directory.user">https://www.googleapis.com/auth/admin.directory.user</a>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.



Exemplary Configuration:

```
ProxyType=Internet

Type=HTTP

Authentication=BasicAuthentication

URL=https://www.googleapis.com/admin/directory

User=1234567890-compute@developer.gserviceaccount.com

Password=-----BEGIN PRIVATE KEY-----\n123ABCDEFG123456789...
... /123456789ABCDEFG123=\n-----END PRIVATE KEY-----\n

OAuth2TokenServiceURL=https://www.googleapis.com/oauth2/v4/token

jwt.subject=john.smith@me123.accounts.ondemand.com

jwt.scope=https://www.googleapis.com/auth/admin.directory.user
```

---

## 6. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [Google G Suite](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

### Caution

An initial password setup is mandatory for all newly provisioned users. This is required by the Google G Suite API and must be provided when new accounts are created. The constant value that you see as configuration for the password attribute in the default transformation is generated by SAP. You have to change the constant value with another one, known only by the representatives of your company, before starting to use the Identity Provisioning service for creating users in your corporate Google G Suite system automatically.

You can change the default transformation mapping rules to reflect your current setup of entities in your Google G Suite. For more information, see:

[Manage Transformations \[page 1494\]](#)

[Google Directory API: Users](#) 

[Google Directory API: Groups](#) 

Default read and write transformations:

### Tip

The proxy [Read Transformation](#) is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a [source](#) one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "sourcePath":
"$$.id",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath":
"$$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$$.meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix":
"${entityIdSourceSystem}"
          }
        ],
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$$.schemas[0]"
      },
      {
        "sourcePath":
"$$.primaryEmail",
        "targetPath":
"$$.emails[0].value",
        "correlationAttribute": true
      },
      {
        "sourcePath":
"$$.primaryEmail",
        "targetPath":
"$$.userName"
      },
      {
        "targetPath":
"$$.emails[0].primary",
        "constant": true
      },
      {
        "sourcePath":
"$$.name",

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "condition":
"($.emails.length() > 0) && ($.name.familyName EMPTY false)",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$$.id"
      },
      {
        "sourcePath":
"$$.name",
        "targetPath":
"$$.name"
      },
      {
        "sourcePath":
"$$.emails[0].value",
        "targetPath":
"$$.primaryEmail"
      },
      {
        "sourcePath":
"$$.phoneNumbers",
        "targetPath":
"$$.phones",
        "optional": true
      },
      {
        "scope":
"createEntity",
        "targetPath":
"$$.password",
        "functions": [
          {
            "type":
"randomPassword",
            "passwordLength": 16,
            "minimumNumberOfLowercaseLetters": 1,
            "minimumNumberOfUppercaseLetters": 1,
            "minimumNumberOfDigits": 1,
            "minimumNumberOfSpecialSymbols": 0
          }
        ],
        "constant":
>false",

```

```

    "targetPath":
    "$.name"
    },
    {
        "constant": true,
        "targetPath":
        "$.active"
    },
    {
        "condition":
        "$.suspended == true",
        "constant": false,
        "targetPath":
        "$.active"
    }
    ],
    "group": {
        "scimEntityEndpoint":
        "Groups",
        "mappings": [
            {
                "constant":
                "urn:ietf:params:scim:schemas:core:2.0:Group",
                "targetPath":
                "$.schemas[0]"
            },
            {
                "sourcePath":
                "$.id",
                "targetVariable":
                "entityIdSourceSystem",
                "targetPath":
                "$.id"
            },
            {
                "sourceVariable":
                "entityBaseLocation",
                "targetVariable":
                "entityLocationSourceSystem",
                "targetPath":
                "$.meta.location",
                "functions": [
                    {
                        "type":
                        "concatString",
                        "suffix":
                        "${entityIdSourceSystem}"
                    }
                ]
            },
            {
                "sourcePath":
                "$.name",
                "targetPath":
                "$.displayName"
            },
            {
                "sourcePath":
                "$.members[?(@.type == 'USER')
                && (@.status == 'ACTIVE')]",

```

```

    "targetPath":
    "$.suspended"
    },
    {
        "condition":
        "$.active == false",
        "constant": true,
        "targetPath":
        "$.suspended"
    },
    {
        "constant":
        "true",
        "targetPath":
        "$.changePasswordAtNextLogin"
    }
    ],
    "group": {
        "scimEntityEndpoint":
        "Groups",
        "mappings": [
            {
                "constant":
                "urn:ietf:params:scim:schemas:core:2.0:Group",
                "targetPath":
                "$.schemas[0]"
            },
            {
                "sourceVariable":
                "entityIdTargetSystem",
                "targetPath":
                "$.id"
            },
            {
                "sourcePath":
                "$.displayName",
                "targetPath":
                "$.email",
                "scope":
                "createEntity"
            },
            {
                "sourcePath":
                "$.displayName",
                "targetPath":
                "$.name"
            },
            {
                "sourcePath":
                "$.members",
                "targetPath":
                "$.members",
                "preserveArrayWithSingleElement":
                true,
                "optional": true
            },
            {
                "targetPath":
                "$.members[*].value",
                "type": "rename",

```

```

    "targetPath":
    "$.members",
    "optional": true,
    "preserveArrayWithSingleElement":
    true
    },
    {
      "targetPath":
      "$.members[*].status",
      "type": "remove"
    },
    {
      "constant":
      "value",
      "targetPath":
      "$.members[*].id",
      "type": "rename"
    },
    {
      "targetPath":
      "$.members[*].kind",
      "type": "remove"
    },
    {
      "targetPath":
      "$.members[*].etag",
      "type": "remove"
    },
    {
      "targetPath":
      "$.members[*].role",
      "type": "remove"
    },
    {
      "constant":
      "display",
      "targetPath":
      "$.members[*].email",
      "type": "rename"
    }
  ]
}

```

```

    "constant": "id"
  },
  {
    "targetPath":
    "$.members[*].display",
    "type": "remove"
  }
]
}

```

If the **displayName** attribute in the source system transformation does not provide group e-mails, you can modify the transformation the following ways:

- Map **email** to another attribute that contains a unique group e-mail.
- Concatenate the **displayName** attribute with your domain. For example:

#### Sample Code

```

{
  "sourcePath": "$.displayName",
  "targetPath": "$.email",
  "scope": "createEntity",
  "functions": [
    {

```

```

        "type": "concatString",
        "suffix": "@test.myaccount.ondemand.com"
      }
    ]
  }

```

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

#### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

#### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

### 1.6.3.37 LDAP Server

Follow this procedure to set up LDAP Server as a proxy system.

## Prerequisites

#### i Note

This system is available for all standalone tenants. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it as a proxy system for reading entities only.

#### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#) or [Cloud Connector \(Cloud Foundry\)](#)
- **For tenants running on the infrastructure of SAP Cloud Identity Services:** You have a multi-environment subaccount in the Cloud Foundry region that maps the region of your Identity Authentication tenant and it is subscribed to the [Cloud Identity Services](#) application. For more information, see [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure](#).
- You have the credentials of a technical user in the LDAP Server, which is used to call the LDAP Server API to read and write users and their attributes.

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context


You can use LDAP Server as a proxy connector to execute [hybrid](#) scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to the LDAP Server, whenever the external back-end requests such.

This scenario supports provisioning **users**, **groups** and **group assignments**.

There are two versions of the LDAP Server connector. Both consume the LDAP Server API to read and write users and groups. The versions are handled by the `ldap.api.version` property as follows:

- When the value is set to [1](#) or the property is not defined (typical for systems created before versioning was introduced on May 25, 2023) LDAP Server API version 1 is used. This is the default value of `ldap.api.version`.  
When using this version of the connector, the entities (users and groups) are read with all attributes. In this version, the `group members` attribute mapping in the proxy read transformation does not include `type` sub-attribute. In this case, all members are considered of type `User`, which is the sub-attribute fallback value. As a consequence, if the external system includes nested groups, they will not be handled properly.
- When the value is set to [2](#) – LDAP Server API version 2 is used.  
This version of the connector comes with improved performance of the read operation for user and group attributes. You are now able to define which user and group attributes to be read. This is possible by adding values to the properties `ldap.user.attributes` or `ldap.group.attributes`.  
Via these properties, you are able to add also user and group operational attributes (attributes which the directory organizes for internal use). For more information, refer to the official LDAP server documentation. After the additional values of the properties are set, the default read or proxy read transformations should also be adjusted accordingly.  
In this version, the `group members` attribute mapping in the proxy read transformation is enhanced with `type` sub-attribute. The sub-attribute has two possible values – `User` and `Group`. This allows you to read and preserve nested groups.  
For more information on how to update to LDAP Server connector version 2, see [Update Connector Version \[page 1484\]](#).

## SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.



Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example: `GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber eq '<attribute>'`
- If your system supports multivalued e-mails (that is `$.mail[0].value`, `$.mail[1].value`, etc.), the search criteria will always resolve only one user e-mail. For LDAP-based systems, this is the first user e-mail (`$.mail[0].value`).

### ❁ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.ldap.attribute.user.id%{0}",
  "targetVariable": "entityIdSourceSystem",
  "targetPath": "$.id",
  "correlationAttribute": true
},
```

Since [LDAP Server](#) supports filtering by userName via the property `ldap.attribute.user.id = cn`, which is set in the [Properties](#) tab.

You also set the following filter in the [Properties](#) tab: `ldap.user.filter = (mail=john.smith03@dummymail.com)`

Thus if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=userName eq "John Smith 03"`

The query request to the LDAP Server API will result into: `(&(cn=John Smith 03)(mail=john.smith03@dummymail.com))`

## Procedure

1. Open Cloud Connector to add an access control system mapping for **LDAP Server**. This is needed to allow the Identity Provisioning service to access LDAP Server as a back-end system on the intranet. To learn how, see: [Configure Access Control \(LDAP\)](#)
2. Depending on the infrastructure of your Identity Provisioning tenant, proceed as follows:
  - **SAP Cloud Identity infrastructure:** Open SAP Cloud Identity Services admin console.
  - **Neo environment:** Open your subaccount in SAP BTP cockpit (valid for OAuth authentication to the Identity Provisioning proxy system).

### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

3. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <i>Users &amp; Authorizations</i> ► <i>Administrators</i> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <i>Access Proxy System API</i> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <i>Security</i> ► <i>OAuth</i> ► <i>Clients</i> ► and choose <i>Register New Client</i>.</li> <li>2. From the <i>Subscription</i> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <i>Authorization Grant</i> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <i>Secret</i> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <i>Client ID</i>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <i>Subscriptions</i> ► <i>Java Applications</i> ► <i>ipsproxy</i> ►.</li> <li>7. From the left-side navigation, choose ► <i>Roles</i> ► <i>IPS_PROXY_USER</i> ►.</li> <li>8. Choose <i>Assign</i> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

4. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
5. Add *LDAP Server* as a proxy system. For more information, see [Add a System \[page 1477\]](#).
6. Choose the *Properties* tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the *Destination Name* combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the *Properties* tab, the value set in the *Properties* tab is considered with higher priority.

We recommend that you use the *Properties* tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">LDAP</a>
ldap.url	Specify the destination URL. It must be in the following format:  ldap://<external_host>:<external_port>
ldap.proxyType	Enter: <a href="#">OnPremise</a>
ldap.authentication	Enter: <a href="#">BasicAuthentication</a>
ldap.user	Enter the <a href="#">distinguishedName</a> of the technical LDAP user. This is the user you need to establish the connection and to perform all queries.
ldap.password	(Credential) Enter the password for the LDAP technical user.
ldap.group.path	Enter the complete path to the node containing the groups in the LDAP tree.
ldap.user.path	Enter the complete path to the users in the LDAP tree.
(Optional)ldap.api.version	Defines the version of LDAP Server API.  <b>Possible values:</b> <ul style="list-style-type: none"> <li>• <b>1</b> - Indicates that LDAP Server API version 1 is used.</li> <li>• <b>2</b> - Indicates that LDAP Server API version 2 is used.</li> </ul> <p>If the property is not defined - LDAP Server API version 1 is used.</p>

### → Remember

We strongly recommend that you enter different paths for LDAP users and groups. That means, the value of `ldap.user.path` should be different than the value of `ldap.group.path`.

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

The LDAP Server proxy system is created by default with the properties listed below:

Default LDAP Properties

```
ldap.user.object.class= inetOrgPerson
ldap.group.object.class= groupOfNames
ldap.group.uniqueattribute= cn
ldap.attribute.group.object.class.required=cn
ldap.attribute.user.object.class.required=cn
ldap.attribute.group.id=cn
ldap.attribute.group.member= member
ldap.attribute.user.id= uid
ldap.attribute.dn=distinguishedName
ldap.attribute.user.mail= mail
ldap.attribute.user.mobile=mobile
ldap.attribute.user.givenName= givenName
ldap.attribute.user.surname= sn
ldap.attribute.user.groups= memberOf
ldap.attribute.user.telephoneNumber= telephoneNumber
ldap.respond.with.resource.after.create=true
ldap.respond.with.resource.after.update=true
```

---

#### **i** Note

The **ldap.attribute.\*** properties are used as parameterized properties in the default transformation. That is, if a property used in the transformation doesn't have a value, the provisioning job will fail when the transformation is loaded on runtime and the property value is substituted.

Also, you can change a property and use a new one (with a new name). In this case, you must replace the old property with the new one at all corresponding places in the transformation.

#### 7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [LDAP Server](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your LDAP Server. For more information, see [Manage Transformations \[page 1494\]](#).

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Default read and write transformations for LDAP Server connector version 1:

### Read Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$.%ldap.attribute.user.id%[0]",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id",
        "correlationAttribute":
true
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$.%meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$.%ldap.attribute.user.id%[0]",
        "targetPath":
"$.%userName",
        "correlationAttribute":
true
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$.%schemas[0]"
      },
      {
        "sourcePath":
"$.%ldap.attribute.user.mail%[0]",
        "targetPath":
"$.%emails[0].value",
        "optional": true,
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$.%ldap.attribute.user.givenName%[0]",
```

### Write Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$.%userName",
        "targetPath":
"$.%ldap.attribute.user.id%[0]",
        "targetVariable":
"entityIdTargetSystem",
        "scope": "createEntity"
      },
      {
        "sourcePath":
"$.%userName",
        "targetPath":
"$.%ldap.attribute.user.object.class.required%[0]"
      },
      {
        "sourcePath":
"$.%emails[*].value",
        "optional": true,
        "targetPath":
"$.%ldap.attribute.user.mail%"
      },
      {
        "sourcePath":
"$.%name.givenName",
        "optional": true,
        "targetPath":
"$.%ldap.attribute.user.givenName%[0]"
      },
      {
        "sourcePath":
"$.%name.familyName",
        "optional": true,
        "targetPath":
"$.%ldap.attribute.user.surname%[0]"
      },
      {
        "sourcePath":
"$.%phoneNumbers[*].value",
        "optional": true,
        "targetPath":
"$.%ldap.attribute.user.mobile%"
      }
    ],
    "group": {
      "scimEntityEndpoint":
"Groups",
      "mappings": [
```

```

        "targetPath":
"$ .name.givenName",
        "optional": true
    },
    {
        "sourcePath":
"$ .%ldap.attribute.user.surname%[0]",
        "targetPath":
"$ .name.familyName",
        "optional": true
    },
    {
        "sourcePath":
"$ .%ldap.attribute.user.groups%",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$ .groups[?
(@.value)]"
    },
    {
        "sourcePath":
"$ .%ldap.attribute.user.mobile%[0]",
        "optional": true,
        "targetPath":
"$ .phoneNumbers[0].value"
    },
    {
        "condition":
"$ .%ldap.attribute.user.mobile%.length() > 0",
        "constant": "mobile",
        "targetPath":
"$ .phoneNumbers[0].type"
    },
    {
        "sourcePath":
"$ .%ldap.attribute.user.telephoneNumber%[0]",
        "optional": true,
        "targetPath":
"$ .phoneNumbers[1].value"
    },
    {
        "condition":
"$ .%ldap.attribute.user.telephoneNumber%.length() > 0",
        "constant": "work",
        "targetPath":
"$ .phoneNumbers[1].type"
    }
]
},
"group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [

```

```

        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .%ldap.attribute.group.id%[0]",
        "targetVariable":
"entityIdTargetSystem",
        "scope": "createEntity"
    },
    {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .%ldap.attribute.group.object.class.required%[0]"
    },
    {
        "constant": [],
        "targetPath": "$ .member"
    },
    {
        "sourcePath":
"$ .members[*]",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetVariable":
"membersVariable",
        "functions": [
            {
                "condition":
"(@.type != 'Group') &&
('%ldap.attribute.user.id%' !=
'%ldap.attribute.dn%')",
                "function":
"concatString",
                "applyOnElements":
true,
                "applyOnAttribute":
"value",
                "prefix":
"%ldap.attribute.user.id%=",
                "suffix":
"%ldap.user.path%"
            },
            {
                "condition":
"(@.type == 'Group') &&
('%ldap.attribute.group.id%' !=
'%ldap.attribute.dn%')",
                "function":
"concatString",
                "applyOnElements":
true,
                "applyOnAttribute":
"value",
                "prefix":
"%ldap.attribute.group.id%=",
                "suffix":
"%ldap.group.path%"
            }
        ]
    },

```

## Read Transformation

```

      "sourcePath":
"$.%ldap.attribute.group.id%[0]",
      "targetVariable":
"entityIdSourceSystem",
      "targetPath": "$.id"
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetVariable":
"entityLocationSourceSystem",
      "targetPath":
"$ .meta.location",
      "functions": [
        {
          "type":
"concatString",
          "suffix": "$
{entityIdSourceSystem}"
        }
      ],
    },
    {
      "sourcePath":
"$.%ldap.attribute.group.id%[0]",
      "targetPath":
"$ .displayName"
    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath":
"$ .schemas[0]"
    },
    {
      "sourcePath":
"$.%ldap.attribute.group.member%",
      "preserveArrayWithSingleElement":
true,
      "targetPath": "$.members[?
(@.value)]",
      "optional": true
    }
  ]
}

```

## Write Transformation

```

    {
      "sourceVariable":
"membersVariable",
      "variablePath": "$
[*].value",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath": "$.member"
    }
  ]
}

```



## Default read transformation for LDAP Server connector version 2:

### Read Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$.%ldap.attribute.user.id%[0]",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id",
        "correlationAttribute":
true
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$..meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$.%ldap.attribute.user.id%[0]",
        "targetPath":
"$..userName",
        "correlationAttribute":
true
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$..schemas[0]"
      },
      {
        "sourcePath":
"$.%ldap.attribute.user.mail%[0]",
        "targetPath":
"$..emails[0].value",
        "optional": true,
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$.%ldap.attribute.user.givenName%[0]",
```

### Write Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$..userName",
        "targetPath":
"$.%ldap.attribute.user.id%[0]",
        "targetVariable":
"entityIdTargetSystem",
        "scope": "createEntity"
      },
      {
        "sourcePath":
"$..userName",
        "targetPath":
"$.%ldap.attribute.user.object.class.required%[0]"
      },
      {
        "sourcePath":
"$..emails[*].value",
        "optional": true,
        "targetPath":
"$.%ldap.attribute.user.mail%"
      },
      {
        "sourcePath":
"$..name.givenName",
        "optional": true,
        "targetPath":
"$.%ldap.attribute.user.givenName%[0]"
      },
      {
        "sourcePath":
"$..name.familyName",
        "optional": true,
        "targetPath":
"$.%ldap.attribute.user.surname%[0]"
      },
      {
        "sourcePath":
"$..phoneNumbers[*].value",
        "optional": true,
        "targetPath":
"$.%ldap.attribute.user.mobile%"
      }
    ],
    "group": {
      "scimEntityEndpoint":
"Groups",
      "mappings": [
```

```

        "targetPath":
"$ .name.givenName",
        "optional": true
    },
    {
        "sourcePath":
"$ .%ldap.attribute.user.surname%[0]",
        "targetPath":
"$ .name.familyName",
        "optional": true
    },
    {
        "sourcePath":
"$ .%ldap.attribute.user.groups%",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetPath": "$ .groups[?
(@.value)]"
    },
    {
        "sourcePath":
"$ .%ldap.attribute.user.mobile%[0]",
        "optional": true,
        "targetPath":
"$ .phoneNumbers[0].value"
    },
    {
        "condition":
"$ .%ldap.attribute.user.mobile%.length() > 0",
        "constant": "mobile",
        "targetPath":
"$ .phoneNumbers[0].type"
    },
    {
        "sourcePath":
"$ .%ldap.attribute.user.telephoneNumber%[0]",
        "optional": true,
        "targetPath":
"$ .phoneNumbers[1].value"
    },
    {
        "condition":
"$ .%ldap.attribute.user.telephoneNumber%.length() > 0",
        "constant": "work",
        "targetPath":
"$ .phoneNumbers[1].type"
    }
]
},
"group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [

```

```

        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .%ldap.attribute.group.id%[0]",
        "targetVariable":
"entityIdTargetSystem",
        "scope": "createEntity"
    },
    {
        "sourcePath":
"$ .displayName",
        "targetPath":
"$ .%ldap.attribute.group.object.class.required%[0]"
    },
    {
        "constant": [],
        "targetPath": "$ .member"
    },
    {
        "sourcePath":
"$ .members[*]",
        "preserveArrayWithSingleElement":
true,
        "optional": true,
        "targetVariable":
"membersVariable",
        "functions": [
            {
                "condition":
"(@.type != 'Group') &&
('%ldap.attribute.user.id%' !=
'%ldap.attribute.dn%')",
                "function":
"concatString",
                "applyOnElements":
true,
                "applyOnAttribute":
"value",
                "prefix":
"%ldap.attribute.user.id%=",
                "suffix":
",%ldap.user.path%"
            },
            {
                "condition":
"(@.type == 'Group') &&
('%ldap.attribute.group.id%' !=
'%ldap.attribute.dn%')",
                "function":
"concatString",
                "applyOnElements":
true,
                "applyOnAttribute":
"value",
                "prefix":
"%ldap.attribute.group.id%=",
                "suffix":
",%ldap.group.path%"
            }
        ]
    },

```

```

      "sourcePath":
"$.%ldap.attribute.group.id%[0]",
      "targetVariable":
"entityIdSourceSystem",
      "targetPath": "$.id"
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetVariable":
"entityLocationSourceSystem",
      "targetPath":
"$ .meta.location",
      "functions": [
        {
          "type":
"concatString",
          "suffix": "$
{entityIdSourceSystem}"
        }
      ],
    },
    {
      "sourcePath":
"$.%ldap.attribute.group.id%[0]",
      "targetPath":
"$ .displayName"
    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath":
"$ .schemas[0]"
    },
    {
      "sourcePath":
"$.%ldap.attribute.group.member%",
      "preserveArrayWithSingleElement":
true,
      "targetPath": "$.members",
      "optional": true
    }
  ]
}

```

```

    {
      "sourceVariable":
"membersVariable",
      "variablePath": "$
[*].value",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath": "$.member"
    }
  ]
}

```

### Note

By default, the **cn** attribute is used for every read or written group. An administrator can change this behavior by setting the following properties:

- `ldap.group.uniquename.attribute` – the value can be either the CN or the whole DN (**distinguishedName**) of the group.
- `ldap.attribute.group.id` – the value can be CN or another attribute to be used as a group ID instead (for example, **displayName** or **description**).

For more information about these properties, see: [List of Properties \[page 94\]](#)

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"><li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li><li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li></ul>	<ul style="list-style-type: none"><li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li><li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li></ul>

#### i Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Related Information

[Technical Documents](#) ➤

[Setting Timeout for Ldap Operations](#) ➤

[Connection Pooling Configuration](#) ➤

## 1.6.3.38 Microsoft Active Directory

Follow this procedure to set up Microsoft Active Directory as a proxy system.

## Prerequisites

#### i Note

If you have purchased the Identity Provisioning service between **September 1, 2020** and **October 20, 2020**, and you want to make a connection to this on-premise system, follow the procedure on page: [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

- You have installed the Cloud Connector in your corporate environment and have done the initial configuration. For more information, see: [Cloud Connector \(Neo\)](#)
- You have the credentials of a technical user in the Microsoft Active Directory, which is used to call the Microsoft Active Directory API to read and write users, groups and their attributes.

### Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

## Context


You can use Microsoft Active Directory as a proxy connector to execute *hybrid* scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to Microsoft Active Directory, whenever the external back-end requests such.

This scenario supports provisioning **users**, **groups** and **group assignments**.

There are two versions of the Microsoft AD connector. Both consume the LDAP Server API to read and write users and groups. The versions are handled by the `ldap.api.version` property as follows:

- When the value is set to **1** or the property is not defined (typical for systems created before versioning was introduced on June 19, 2023) LDAP Server API version 1 is used. This is the default value of `ldap.api.version`.  
When using this version of the connector, the entities (users and groups) are read with all attributes. In this version, the `group members` attribute mapping in the proxy read transformation does not include `type` sub-attribute. In this case, all members are considered of type `User`, which is the sub-attribute fallback value. As a consequence, if the external system includes nested groups, they will not be handled properly.
- When the value is set to **2** – LDAP Server API version 2 is used.  
This version of the connector is with improved performance of the read operation for user and group attributes. You are now able to define which user and group attributes to be read. This is possible by adding values to the properties `ldap.user.attributes` or `ldap.group.attributes`.  
Via these properties, you are able to add also user and group operational attributes (attributes which the directory organizes for internal use). For more information, refer to the official LDAP server documentation. After the additional values of the properties are set, the default read or proxy read transformations should also be adjusted accordingly.  
In this version, the `group members` attribute mapping in the proxy read transformation is enhanced with `type` sub-attribute. The sub-attribute has two possible values – `User` and `Group`. This allows you to read and preserve nested groups.  
For more information on how to update to Microsoft Active Directory version 2, see [Update Connector Version \[page 1484\]](#).

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example: *GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber eq '<attribute>'*
- If your system supports multivalued e-mails (that is *\$.mail[0].value*, *\$.mail[1].value*, etc.), the search criteria will always resolve only one user e-mail. For LDAP-based systems, this is the first user e-mail (*\$.mail[0].value*).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.sAMAccountName[0]",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `ldap.user.filter = (mail=john.smith03@dummymail.com)`

Thus If, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "jSmith03"**

The query request to the Microsoft Active Directory API will result into: **(&(sAMAccountName[0]=jSmith03)(mail=john.smith03@dummymail.com))**

## Procedure

1. Open Cloud Connector to add an access control system mapping for **Microsoft Active Directory**. This is needed to allow the Identity Provisioning service to access Microsoft Active Directory as a back-end system on the intranet. To learn how, see: [Configure Access Control \(LDAP\)](#)
2. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit ► [Neo](#) ► [Overview](#) , you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

3. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

4. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
5. Add [Microsoft Active Directory](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
6. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">LDAP</a>
ldap.url	Specify a destination URL. It must be in the following format:  ldap://<ext_host>:<ext_port>
ldap.proxyType	Enter: <a href="#">OnPremise</a>
ldap.authentication	Enter: <a href="#">BasicAuthentication</a>
ldap.user	Enter the <a href="#">distinguishedName</a> or the <a href="#">userPrincipalName</a> of the Microsoft AD technical user. This is the user you need to establish the connection and to perform all queries.
ldap.password	(Credential) Enter the password for the Microsoft AD technical user.
ldap.attribute.user.id	Default property, which denotes the ID of a user.  By default, it's set to: <a href="#">cn</a>
ldap.attribute.group.id	Default property, which denotes the ID of a group.  By default, it's set to: <a href="#">cn</a>
ldap.attribute.dn	Default property, which denotes the distinguished name of a user or a group.  Only possible value: <a href="#">distinguishedName</a>
ldap.respond.with.resource.after.create	Default property, whose value is <a href="#">true</a> .
ldap.respond.with.resource.after.update	Default property, whose value is <a href="#">true</a> .
ldap.group.path	Enter the complete path to the node containing the groups in Microsoft Active Directory.
ldap.user.path	Enter the complete path to the users in Microsoft Active Directory.
(Optional)ldap.api.version	Handles the version of the LDAP Server connector.  <b>Possible values:</b> <ul style="list-style-type: none"> <li>• <a href="#">1</a> - Indicates that Microsoft AD API version 1 is used.</li> <li>• <a href="#">2</a> - Indicates that Microsoft AD API version 2 is used.</li> </ul> Default value: <a href="#">1</a>



Example for a destination or a set of properties:

```
Type=LDAP

Name=MyADDestination

ldap.user=john.smith@some.dummy.domain.com

ldap.password=*****

ldap.attribute.user.id=cn

ldap.attribute.group.id=cn

ldap.attribute.dn=distinguishedName

ldap.url=ldap://abcd:123

ldap.proxyType=OnPremise

ldap.authentication=BasicAuthentication

ldap.group.path=OU=Groups,OU=IAS,DC=global,DC=corp,DC=mycompany

ldap.user.path=OU=Users,OU=IAS,DC=global,DC=corp,DC=mycompany
```

---

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 7. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *Microsoft Active Directory* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your Microsoft AD. For more information, see:

[Manage Transformations \[page 1494\]](#)

[MS Graph: Users](#) ➡

[MS Graph: Groups](#) ➡

Default read and write transformations:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users* or */Groups*) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Default read and write transformations for Microsoft AD connector version 1:

### Read Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$.%ldap.attribute.user.id%",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id",
        "correlationAttribute":
true
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$ .sAMAccountName%",
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$ .displayName%",
        "targetPath":
"$ .displayName",
        "optional": true
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "sourcePath": "$ .mail%",
        "targetPath":
"$ .emails[0].value",
        "optional": true,
```

### Write Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "condition":
"('%ldap.attribute.user.id%' !=
'%ldap.attribute.dn%')",
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .%ldap.attribute.user.id%",
        "targetVariable":
"entityIdTargetSystem",
        "scope": "createEntity"
      },
      {
        "condition":
"// If a user is not a direct
member of the configured user
base path, then its
distinguishedName is configured
to be equal to CN =
<userName>,<nested_path>,<base_pat
h>,
// where <nested_path> is read
from \"sourcePath\": \"$
['urn:sap:cloud:scim:schemas:exten
sion:ad:2.0:User']['nestedPath']"
      },
      {
        "condition":
"('%ldap.attribute.user.id%' ==
'%ldap.attribute.dn%')",
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:exten
sion:ad:2.0:User']['nestedPath']",
        "optional": true,
        "targetVariable":
"nestedPathVariable",
        "defaultValue": "",
        "scope": "createEntity"
      },
      {
        "condition":
"('%ldap.attribute.user.id%' ==
'%ldap.attribute.dn%')",
        "sourcePath":
"$ .userName",
        "functions": [
          {
            "function":
"concatString",
            "prefix": "CN="
          },
          {
            "condition": "('$
{nestedPathVariable}' != '')",
```

```

        "correlationAttribute":
true
        {
            "sourcePath":
"$$.givenName[0]",
            "targetPath":
"$$.name.givenName",
            "optional": true
        },
        {
            "sourcePath": "$$.sn[0]",
            "targetPath":
"$$.name.familyName",
            "optional": true
        },
        {
            "sourcePath":
"$$.memberOf",
            "preserveArrayWithSingleElement":
true,
            "optional": true,
            "targetPath": "$$.groups[?
(@.value)]"
        },
        {
            "sourcePath":
"$$.mobile[0]",
            "optional": true,
            "targetPath":
"$$.phoneNumbers[0].value"
        },
        {
            "condition":
"$$.mobile.length() > 0",
            "constant": "mobile",
            "targetPath":
"$$.phoneNumbers[0].type"
        },
        {
            "sourcePath":
"$$.telephoneNumber[0]",
            "optional": true,
            "targetPath":
"$$.phoneNumbers[1].value"
        },
        {
            "condition":
"$$.telephoneNumber.length() > 0",
            "constant": "work",
            "targetPath":
"$$.phoneNumbers[1].type"
        }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [

```

```

        "function":
"concatString",
        "suffix": ", ",
        {
            "condition": "('$
{nestedPathVariable}' != '')",
            "function":
"concatString",
            "suffix": "$
{nestedPathVariable}"
        },
        {
            "function":
"concatString",
            "suffix":
",%ldap.user.path%"
        }
    ],
    "targetPath":
"$$.%ldap.attribute.user.id[0]",
    "targetVariable":
"entityIdTargetSystem",
    "scope": "createEntity"
},
{
    "sourcePath":
"$$.userName",
    "targetPath": "$$.cn[0]",
    "scope": "createEntity"
},
{
    "sourcePath":
"$$.userName",
    "targetPath":
"$$.sAMAccountName[0]"
},
{
    "sourcePath":
"$$.displayName",
    "optional": true,
    "targetPath":
"$$.displayName[0]"
},
{
    "sourcePath":
"$$.emails[0].value",
    "optional": true,
    "targetPath": "$$.mail[0]"
},
{
    "sourcePath":
"$$.name.givenName",
    "optional": true,
    "targetPath":
"$$.givenName[0]"
},
{
    "sourcePath":
"$$.name.familyName",
    "optional": true,
    "targetPath": "$$.sn[0]"
}

```

```

      "sourcePath":
"$.%ldap.attribute.group.id%[0]",
      "targetVariable":
"entityIdSourceSystem",
      "targetPath": "$.id"
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetVariable":
"entityLocationSourceSystem",
      "targetPath":
"$ .meta.location",
      "functions": [
        {
          "type":
"concatString",
          "suffix": "$
{entityIdSourceSystem}"
        }
      ],
    },
    {
      "sourcePath":
"$ .sAMAccountName[0]",
      "targetPath":
"$ .displayName"
    },
    {
      "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
      "targetPath":
"$ .schemas[0]"
    },
    {
      "sourcePath": "$.member",
      "preserveArrayWithSingleElement":
true,
      "targetPath": "$.members[?
(@.value)]",
      "optional": true
    }
  ]
}

```

```

    ],
    "group": {
      "scimEntityEndpoint":
"Groups",
      "mappings": [
        {
          "condition":
"('%ldap.attribute.group.id%' !=
'%ldap.attribute.dn%')",
          "sourcePath":
"$ .displayName",
          "targetPath":
"$.%ldap.attribute.group.id%[0]",
          "targetVariable":
"entityIdTargetSystem",
          "scope": "createEntity"
        },

        // If a group is not a direct
        // member of the configured group
        // base path, then its
        // distinguishedName is configured
        // to be equal to CN =
        // <displayName>,<nested_path>,<base_
        // path>,
        // where <nested_path> is read
        // from "sourcePath": "$
        // ['urn:sap:cloud:scim:schemas:exten
        // sion:ad:2.0:Group']['nestedPath']"

        {
          "condition":
"('%ldap.attribute.group.id%' ==
'%ldap.attribute.dn%')",
          "sourcePath": "$
['urn:sap:cloud:scim:schemas:exten
sion:ad:2.0:Group']
['nestedPath']",
          "optional": true,
          "targetVariable":
"nestedPathVariable",
          "defaultValue": "",
          "scope": "createEntity"
        },
        {
          "condition":
"('%ldap.attribute.group.id%' ==
'%ldap.attribute.dn%')",
          "sourcePath":
"$ .displayName",
          "functions": [
            {
              "function":
"concatString",
              "prefix": "CN="
            },
            {
              "condition": "('$
{nestedPathVariable}' != '')",
              "function":
"concatString",

```

```

        "suffix": ", "
      },
      {
        "condition": "('$
{nestedPathVariable}' != '')",
        "function":
"concatString",
        "suffix": "$
{nestedPathVariable}"
      },
      {
        "function":
"concatString",
        "suffix":
",%ldap.group.path%"
      }
    ],
    "targetPath":
"$.%ldap.attribute.group.id%[0]",
    "targetVariable":
"entityIdTargetSystem",
    "scope": "createEntity"
  },
  {
    "sourcePath":
"$.displayName",
    "targetPath": "$.cn[0]",
    "scope": "createEntity"
  },
  {
    "sourcePath":
"$.displayName",
    "targetPath":
"$.$sAMAccountName[0]"
  },
  {
    "constant": [],
    "targetPath": "$.member"
  },
  {
    "sourcePath":
"$.$members[*]",
    "preserveArrayWithSingleElement":
true,
    "optional": true,
    "targetVariable":
"membersVariable",
    "functions": [
      {
        "condition":
"(@.type != 'Group') &&
('%ldap.attribute.user.id%' !=
'%ldap.attribute.dn%')",
        "function":
"concatString",
        "applyOnElements":
true,
        "applyOnAttribute":
"value",
        "prefix":
"%ldap.attribute.user.id%=",

```

```
        "suffix":
        ",%ldap.user.path%"
    },
    {
        "condition":
        "(@.type == 'Group') &&
        ('%ldap.attribute.group.id%' !=
        '%ldap.attribute.dn%')",
        "function":
        "concatString",
        "applyOnElements":
        true,
        "applyOnAttribute":
        "value",
        "prefix":
        "%ldap.attribute.group.id%=",
        "suffix":
        ",%ldap.group.path%"
    }
]
},
{
    "sourceVariable":
    "membersVariable",

    "preserveArrayWithSingleElement":
    true,
    "optional": true,
    "targetPath": "$.member",
    "variablePath": "$
[*].value"
}
]
```

## Default read transformation for Microsoft AD connector version 2:

### Read Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath":
"$.%ldap.attribute.user.id%",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id",
        "correlationAttribute":
true
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath":
"$ .sAMAccountName%",
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$ .displayName%",
        "targetPath":
"$ .displayName",
        "optional": true
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "sourcePath": "$ .mail%",
        "targetPath":
"$ .emails[0].value",
        "optional": true,

```

### Write Transformation

#### Code Syntax

```
{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "condition":
"('%ldap.attribute.user.id%' !=
'%ldap.attribute.dn%')",
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .%ldap.attribute.user.id%",
        "targetVariable":
"entityIdTargetSystem",
        "scope": "createEntity"
      },
      {
        "condition":
"// If a user is not a direct
member of the configured user
base path, then its
distinguishedName is configured
to be equal to CN =
<userName>,<nested_path>,<base_pat
h>,
// where <nested_path> is read
from \"sourcePath\": \"$
['urn:sap:cloud:scim:schemas:exten
sion:ad:2.0:User']['nestedPath']"
      },
      {
        "condition":
"('%ldap.attribute.user.id%' ==
'%ldap.attribute.dn%')",
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:exten
sion:ad:2.0:User']['nestedPath']",
        "optional": true,
        "targetVariable":
"nestedPathVariable",
        "defaultValue": "",
        "scope": "createEntity"
      },
      {
        "condition":
"('%ldap.attribute.user.id%' ==
'%ldap.attribute.dn%')",
        "sourcePath":
"$ .userName",
        "functions": [
          {
            "function":
"concatString",
            "prefix": "CN="
          },
          {
            "condition": "('$
{nestedPathVariable}' != '')",

```



```

        "correlationAttribute":
true
        {
            "sourcePath":
"$$.givenName[0]",
            "targetPath":
"$$.name.givenName",
            "optional": true
        },
        {
            "sourcePath": "$$.sn[0]",
            "targetPath":
"$$.name.familyName",
            "optional": true
        },
        {
            "sourcePath":
"$$.memberOf",
            "targetPath":
"$$.preserveArrayWithSingleElement":
true,
            "optional": true,
            "targetPath": "$$.groups[?
(@.value)]"
        },
        {
            "sourcePath":
"$$.mobile[0]",
            "optional": true,
            "targetPath":
"$$.phoneNumbers[0].value"
        },
        {
            "condition":
"$$.mobile.length() > 0",
            "constant": "mobile",
            "targetPath":
"$$.phoneNumbers[0].type"
        },
        {
            "sourcePath":
"$$.telephoneNumber[0]",
            "optional": true,
            "targetPath":
"$$.phoneNumbers[1].value"
        },
        {
            "condition":
"$$.telephoneNumber.length() > 0",
            "constant": "work",
            "targetPath":
"$$.phoneNumbers[1].type"
        }
    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [

```

```

        "function":
"concatString",
        "suffix": ", ",
        {
            "condition": "('$
{nestedPathVariable}' != '')",
            "function":
"concatString",
            "suffix": "$
{nestedPathVariable}"
        },
        {
            "function":
"concatString",
            "suffix":
",%ldap.user.path%"
        }
    ],
    "targetPath":
"$$.%ldap.attribute.user.id[0]",
    "targetVariable":
"entityIdTargetSystem",
    "scope": "createEntity"
},
{
    "sourcePath":
"$$.userName",
    "targetPath": "$$.cn[0]",
    "scope": "createEntity"
},
{
    "sourcePath":
"$$.userName",
    "targetPath":
"$$.sAMAccountName[0]"
},
{
    "sourcePath":
"$$.displayName",
    "optional": true,
    "targetPath":
"$$.displayName[0]"
},
{
    "sourcePath":
"$$.emails[0].value",
    "optional": true,
    "targetPath": "$$.mail[0]"
},
{
    "sourcePath":
"$$.name.givenName",
    "optional": true,
    "targetPath":
"$$.givenName[0]"
},
{
    "sourcePath":
"$$.name.familyName",
    "optional": true,
    "targetPath": "$$.sn[0]"
}

```

```

        "sourcePath":
"$.%ldap.attribute.group.id%[0]",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
    },
    {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
            {
                "type":
"concatString",
                "suffix": "$
{entityIdSourceSystem}"
            }
        ],
        {
            "sourcePath":
"$ .sAMAccountName[0]",
            "targetPath":
"$ .displayName"
        },
        {
            "constant":
"urn:ietf:params:scim:schemas:core:2.0:Group",
            "targetPath":
"$ .schemas[0]"
        },
        {
            "sourcePath": "$.member",
            "targetPath": "$.members",
            "optional": true
        }
    ]
}
}

```

```

    ],
    "group": {
        "scimEntityEndpoint":
"Groups",
        "mappings": [
            {
                "condition":
"('%ldap.attribute.group.id%' != '%ldap.attribute.dn%')",
                "sourcePath":
"$ .displayName",
                "targetPath":
"$ .%ldap.attribute.group.id%[0]",
                "targetVariable":
"entityIdTargetSystem",
                "scope": "createEntity"
            },

            // If a group is not a direct
            // member of the configured group
            // base path, then its
            // distinguishedName is configured
            // to be equal to CN =
            // <displayName>, <nested_path>, <base_
            // path>,
            // where <nested_path> is read
            // from "sourcePath": "$
            // ['urn:sap:cloud:scim:schemas:exten
            // sion:ad:2.0:Group']['nestedPath']"

            {
                "condition":
"('%ldap.attribute.group.id%' == '%ldap.attribute.dn%')",
                "sourcePath": "$
['urn:sap:cloud:scim:schemas:exten
sion:ad:2.0:Group']['nestedPath']",
                "optional": true,
                "targetVariable":
"nestedPathVariable",
                "defaultValue": "",
                "scope": "createEntity"
            },
            {
                "condition":
"('%ldap.attribute.group.id%' == '%ldap.attribute.dn%')",
                "sourcePath":
"$ .displayName",
                "functions": [
                    {
                        "function":
"concatString",
                        "prefix": "CN="
                    },
                    {
                        "condition": "('$
{nestedPathVariable}' != '')",
                        "function":
"concatString",

```

```

        "suffix": " ",
        },
        {
            "condition": "('$
{nestedPathVariable}' != ' ')",
            "function":
            "concatString",
            "suffix": "$
{nestedPathVariable}"
        },
        {
            "function":
            "concatString",
            "suffix":
            ",%ldap.group.path%"
        }
    ],
    "targetPath":
    "$.%ldap.attribute.group.id%[0]",
    "targetVariable":
    "entityIdTargetSystem",
    "scope": "createEntity"
    },
    {
        "sourcePath":
        "$.displayName",
        "targetPath": "$.cn[0]",
        "scope": "createEntity"
    },
    {
        "sourcePath":
        "$.displayName",
        "targetPath":
        "$.sAMAccountName[0]"
    },
    {
        "constant": [],
        "targetPath": "$.member"
    },
    {
        "sourcePath":
        "$.members[*]",
        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetVariable":
        "membersVariable",
        "functions": [
            {
                "condition":
                "(@.type != 'Group') &&
                ('%ldap.attribute.user.id%' !=
                '%ldap.attribute.dn%')",
                "function":
                "concatString",
                "applyOnElements":
                true,
                "applyOnAttribute":
                "value",
                "prefix":
                "%ldap.attribute.user.id%=",
            }
        ]
    }

```

```

        "suffix":
        ",%ldap.user.path%"
        },
        {
            "condition":
            "(@.type == 'Group') &&
            ('%ldap.attribute.group.id%' !=
            '%ldap.attribute.dn%')",
            "function":
            "concatString",
            "applyOnElements":
            true,
            "applyOnAttribute":
            "value",
            "prefix":
            "%ldap.attribute.group.id%=",
            "suffix":
            ",%ldap.group.path%"
        }
    ],
    {
        "sourceVariable":
        "membersVariable",

        "preserveArrayWithSingleElement":
        true,
        "optional": true,
        "targetPath": "$.member",
        "variablePath": "$
        [*].value"
    }
    ]
}

```

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[Microsoft AD: Technical Documents](#) 




[Setting Timeout for Ldap Operations](#) 

[Connection Pooling Configuration](#) 


## 1.6.3.39 Microsoft Azure Active Directory

Follow this procedure to set up Microsoft Azure Active Directory as a proxy system.

### Prerequisites

- You've logged on to Microsoft Azure Portal, with credentials for a user with directory role **Global administrator**. For more information, see [Microsoft: Assigning administrator roles in Azure Active Directory](#) .
- In ► [Azure Active Directory](#) ► [App registrations](#) ▾, you've registered an application with a secret key and permissions for Microsoft Graph API. These permissions must be consented by an administrator. For more information, see [Microsoft Graph permissions reference](#) .
- (Relevant to target systems) Your registered application is assigned the **User Account Administrator** role. This role allows you to deprovision users. For more information, see [MS Azure PowerShell: Add-MsolRole Member](#) .

#### i Note

If this role isn't assigned, you can only disable users. To do that, set the `accountEnabled` property to **false**. For more information, see [MS Graph: user resource type](#) .

#### i Note

Administrators of bundle tenants on Neo environment should enable the *Manage OAuth Clients* permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).


### Permissions

Assign the following permissions to your application, according to your scenario. Also, the permissions have to be of type *Application*.

- Users – *User.ReadWrite.All, Directory.AccessAsUser.All*
- Groups – *Group.ReadWrite.All*


For more information, see [MS Graph: Users](#)  and [MS Graph: Groups](#) .

### Context

When using it as a proxy system, you can write both users and groups, read from any source system you've added in the Identity Provisioning user interface. The Azure AD target systems use Microsoft Graph API. For more information, see [Microsoft Graph](#) .

If you've successfully finished with the initial setup (described in the **Prerequisites** section), continue with the procedure.

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names ([<schema>:<attribute>](#)) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

### Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userPrincipalName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `aad.user.filter = Department eq 'Finance'`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=username eq "johnsmith@domain.onmicrosoft.com"`

The query request to the Microsoft Graph API will result into: `/User?$filter=Department eq 'Finance' and userPrincipalName eq 'johnsmith@domain.onmicrosoft.com'`

## Procedure

1. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit  [Neo](#)  [Overview](#) , you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you

can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

2. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

SAP Cloud Identity Infrastructure	Neo Environment
For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP Cloud Identity Infrastructure</i>	For <b>Certificate-based authentication</b> , follow the procedure in <a href="#">Manage Certificates for Inbound Connection [page 1510]</a> → <i>SAP BTP, Neo Environment</i>
For <b>Basic authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. In SAP Cloud Identity Services admin console, navigate to ► <a href="#">Users &amp; Authorizations</a> ► <a href="#">Administrators</a> ►.</li> <li>2. Add an administrator user of type <b>System</b> and configure the basic authentication method for this user. If you already have a technical user, skip this step.</li> <li>3. Save your changes.</li> <li>4. Select your administrator user of type <b>System</b> and enable the <a href="#">Access Proxy System API</a> permission.</li> <li>5. Save your changes.</li> </ol>	For <b>OAuth authentication</b> , proceed as follows: <ol style="list-style-type: none"> <li>1. Go to ► <a href="#">Security</a> ► <a href="#">OAuth</a> ► <a href="#">Clients</a> ► and choose <a href="#">Register New Client</a>.</li> <li>2. From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</li> <li>3. From the <a href="#">Authorization Grant</a> combo box, select <b>Client Credentials</b>.</li> <li>4. In the <a href="#">Secret</a> field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.</li> <li>5. Copy/paste and save (in a notepad) the generated <a href="#">Client ID</a>. You will need it later, too.</li> <li>6. From the left-side navigation, choose ► <a href="#">Subscriptions</a> ► <a href="#">Java Applications</a> ► <a href="#">ipsproxy</a> ►.</li> <li>7. From the left-side navigation, choose ► <a href="#">Roles</a> ► <a href="#">IPS_PROXY_USER</a> ►.</li> <li>8. Choose <a href="#">Assign</a> and enter <b>oauth_client_&lt;client_ID&gt;</b>. For <b>&lt;client_ID&gt;</b>, enter the one you have saved in the previous main step.</li> </ol>

3. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
4. Add [Microsoft Azure Active Directory](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
5. Choose the [Properties](#) tab to configure the connection settings for your system.

#### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.



We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

#### Mandatory Properties

Property Name	Description & Value
Type	Enter: <a href="#">HTTP</a>
URL	Enter: <a href="https://graph.microsoft.com">https://graph.microsoft.com</a>
ProxyType	Enter: <a href="#">Internet</a>
Authentication	Enter: <a href="#">BasicAuthentication</a>
User	Enter the application ID registered in your Azure AD subscription (see the <b>Prerequisites</b> section).
Password	Enter the secret key associated to your app registration.
aad.domain.name	Enter one of the verified domain names from the corresponding Azure AD tenant. On this domain, you perform the provisioning operations. For more information, see <a href="#">Microsoft: Manage domain names</a> .
oauth.resource.name	Enter: <a href="https://graph.microsoft.com">https://graph.microsoft.com</a>
OAuth2TokenServiceURL	Enter: <a href="https://login.microsoftonline.com/&lt;your_domain&gt;/oauth2/token">https://login.microsoftonline.com/&lt;your_domain&gt;/oauth2/token</a> , where <your_domain> is the domain name you have set in the <a href="#">aad.domain.name</a> property.

(Optional) `aad.group.member.attributes`

This property defines the attributes of a group member to be read by the Identity Provisioning. By default, it always reads the **type** and the **id** of a member.

If you want the Identity Provisioning to read additional attributes, enter them as a single or a comma-separated value. For example:

#### ❖ Example

- If you want to read the e-mails too, enter:  
`aad.group.member.attributes=mail`  
This will read a member's type, ID, and e-mail.
- If you want to read multiple additional attributes, enter:  
`aad.group.member.attributes=mail,mobilePhone.displayName`  
This will read a member's type, ID, e-mail, phone, and display name.

Property Name	Description & Value
(Optional) <code>aad.user.attributes.membership.active</code>	<p>Use this property if you want to get information about all the groups to which the users are assigned (if any).</p> <ul style="list-style-type: none"> <li>If the property is missing, or is set to <i>false</i> – group membership details for the users will not be extracted.</li> <li>If the property is set to <i>true</i> – group membership details for the users will be extracted.</li> </ul> <p>To learn more, see: <a href="#">List of Properties [page 94]</a></p>
(Optional) <code>aad.user.filter</code>	<p>Use this property to filter users by specific criteria, according to the <a href="#">Microsoft Graph REST API</a> .</p> <div> <p><b>Note</b></p> <p>This property replaces the deprecated <code>msgraph-filter</code> property. To learn more, see: <a href="#">List of Properties [page 94]</a></p> </div>
(Optional) <code>aad.group.filter</code>	<p>Use this property to filter groups by specific criteria, according to the <a href="#">Microsoft Graph REST API</a> .</p>
(Optional) <code>aad.user.filter.group.filter.combine</code>	<p>Use this property to filter users based on their group assignments.</p> <p>When set to <b>true</b>, this property combines user and group filters defined on the <code>aad.user.filter</code> and <code>aad.group.filter</code> properties to further narrow the search results. This way, only users that meet the following filtering criteria are returned:</p> <ul style="list-style-type: none"> <li>Users that match the user filter and at the same time are members of groups that match the group filter.</li> <li>Members of the filtered groups that match the user filter.</li> </ul> <p>When set to <b>false</b>, user and group filters are not combined.</p> <p>To learn more, see: <a href="#">List of Properties [page 94]</a></p>

Property Name	Description & Value
(Optional) <code>aad.user.attributes</code>	<p>Defines which user attributes are read from Microsoft Azure AD system.</p> <p>The property is set during system creation with the following default value:  <a href="#"><i>id,mail,userPrincipalName,displayName,mailNickname,givenName,surname,mobilePhone,businessPhones</i></a></p> <p>This means that by default, Identity Provisioning will read from MS Azure AD the user attributes defined in the property value. Those attributes are used in the default read transformation.</p> <p>To check the complete set of user attributes (properties) supported by Microsoft Azure AD, see: <a href="#">Microsoft Graph: User Properties</a> ➡</p> <p>To learn more, see: <a href="#">List of Properties [page 94]</a></p>
(Optional) <code>aad.group.attributes</code>	<p>Defines which group attributes are read from Microsoft Azure AD system.</p> <p>The property is set during system creation with the following default value: <a href="#"><i>id,displayName,mailNickname</i></a></p> <p>This means that by default, Identity Provisioning will read from MS Azure AD the group attributes defined in the property value and will also return the <a href="#"><i>members</i></a> attribute. Those attributes are used in the default read transformation.</p> <p>To check the complete set of group attributes (properties) supported by Microsoft Azure AD, see: <a href="#">Microsoft Graph: Group Properties</a> ➡</p> <p>To learn more, see: <a href="#">List of Properties [page 94]</a></p>
(Optional) <code>aad.entities.top</code>	<p>This property defines the number of entities to be read per page. Default value: <i>100</i></p>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the [Name](#) or [System Type](#) columns.

#### 6. (Optional) Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the [Microsoft Azure Active Directory](#) proxy system, whose settings are displayed under the [Transformations](#) tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your Microsoft Azure AD. For more information, see:

[Manage Transformations \[page 1494\]](#)

[MS Graph: Users](#) ➡

[MS Graph: Groups](#) ➡

Default read and write transformations:

#### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (**/Users** or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "condition":
"$ .userPrincipalName EMPTY false",
    "mappings": [
      {
        "sourcePath": "$ .id",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$ .id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath":
"$ .schemas[0]"
      },
      {
        "sourcePath": "$ .mail",
        "targetPath":
"$ .emails[0].value",
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$ .userPrincipalName",
        "targetPath":
"$ .userName",
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$ .displayName",
        "optional": true,
        "targetPath":
"$ .displayName"
      },

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
"Users",
    "mappings": [
      {
        "optional": true,
        "sourcePath":
"$ .onPremisesImmutableId",
        "targetPath":
"$ .onPremisesImmutableId"
      },
      {
        "sourcePath":
"$ .active",
        "optional": true,
        "targetPath":
"$ .accountEnabled"
      },
      {
        "sourcePath":
"$ .name.givenName",
        "optional": true,
        "targetPath":
"$ .mailNickname"
      },
      {
        "sourcePath":
"$ .displayName",
        "optional": true,
        "targetPath":
"$ .displayName"
      },
      {
        "sourcePath":
"$ .name.givenName",
        "optional": true,
        "targetPath":
"$ .givenName"
      },
      {
        "sourcePath":
"$ .name.familyName",
        "optional": true,
        "targetPath":
"$ .surname"
      },
      {
        "sourcePath":
"$ .addresses[0].locality",
        "optional": true,
        "targetPath":
"$ .city"
      },
      {
        "sourcePath":
"$ .addresses[0].country",
        "optional": true,
        "targetPath":
"$ .country"
      }
    ]
  }
}

```

```

      "sourcePath":
"$mailNickname",
      "optional": true,
      "targetPath":
"$externalId",
      "correlationAttribute":
true
    },
    {
      "sourcePath":
"$givenName",
      "optional": true,
      "targetPath":
"$name.givenName"
    },
    {
      "sourcePath": "$surname",
      "optional": true,
      "targetPath":
"$name.familyName"
    },
    {
      "sourcePath":
"$mobilePhone",
      "optional": true,
      "targetPath":
"$phoneNumbers[0].value"
    },
    {
      "condition":
"$mobilePhone EMPTY false",
      "constant": "mobile",
      "targetPath":
"$phoneNumbers[0].type"
    },
    {
      "sourcePath":
"$businessPhones[0]",
      "optional": true,
      "targetPath":
"$phoneNumbers[1].value"
    },
    {
      "condition":
"$businessPhones.length() > 0",
      "constant": "work",
      "targetPath":
"$phoneNumbers[1].type"
    },
    {
      "sourcePath": "$groups",
      "preserveArrayWithSingleElement":
true,
      "optional": true,
      "targetPath": "$groups"
    },
    {
      "sourcePath":
"$manager.id",
      "targetPath": "$
[ 'urn:ietf:params:scim:schemas:ext

```

```

    },
    {
      "scope":
"createEntity",
      "sourcePath":
"$userName",
      "targetPath":
"$userPrincipalName",
      "functions": [
        {
          "type":
"concatString",
          "suffix":
"@aad.domain.name%"
        }
      ]
    },
    {
      "scope":
"createEntity",
      "constant":
true,
      "targetPath":
"$accountEnabled"
    },
    {
      "scope":
"createEntity",
      "sourcePath":
"$active",
      "optional": true,
      "targetPath":
"$accountEnabled"
    },
    {
      "scope":
"createEntity",
      "sourcePath":
"name.givenName",
      "targetPath":
"$mailNickname"
    },
    {
      "scope":
"createEntity",
      "sourcePath":
"$displayName",
      "targetPath":
"$displayName"
    },
    {
      "scope":
"createEntity",
      "targetPath":
"$passwordProfile.password",
      "functions": [
        {
          "type":
"randomPassword",
          "passwordLength": 16,

```

```

extension:enterprise:2.0:User']
['manager']['value']",
    "optional": true
  },
  {
    "sourcePath":
"$manager.displayName",
    "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['manager']['displayName']",
    "optional": true
  }
]
},
"group": {
  "scimEntityEndpoint":
"Groups",
  "mappings": [
    {
      "constant":
"urn:ietf:params:scim:schemas:core
:2.0:Group",
      "targetPath":
"$schemas[0]"
    },
    {
      "sourcePath": "$id",
      "targetVariable":
"entityIdSourceSystem",
      "targetPath": "$id"
    },
    {
      "sourceVariable":
"entityBaseLocation",
      "targetVariable":
"entityLocationSourceSystem",
      "targetPath":
"$meta.location",
      "functions": [
        {
          "type":
"concatString",
          "suffix": "$
{entityIdSourceSystem}"
        }
      ]
    },
    {
      "sourcePath":
"$displayName",
      "targetPath":
"$displayName"
    },
    {
      "sourcePath": "$members",
      "targetPath":
"$members"
    }
  ],
  "preserveArrayWithSingleElement":
true,
  "optional": true,
  "targetPath": "$members"
}
}

```

```

"minimumNumberOfLowercaseLetters":
1,
"minimumNumberOfUppercaseLetters":
1,
"minimumNumberOfDigits": 1,
"minimumNumberOfSpecialSymbols": 0
}
],
{
  "scope":
"createEntity",
  "constant": false,
  "targetPath":
"$passwordProfile.forceChangePass
wordNextSignIn"
},
{
  "group": {
    "scimEntityEndpoint":
"Groups",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath":
"$id"
      },
      {
        "optional": true,
        "sourcePath":
"$displayName",
        "targetPath":
"$displayName"
      },
      {
        "scope":
"createEntity",
        "sourcePath":
"$displayName",
        "targetPath":
"$displayName"
      },
      {
        "scope":
"createEntity",
        "sourcePath":
"$externalId",
        "targetPath":
"$mailNickname"
      },
      {
        "scope":
"createEntity",
        "constant": true,
        "targetPath":
"$mailEnabled"
      }
    ]
  }
}

```

#### Read Transformation

```
{
  "targetPath":
"$members[*].id",
  "constant": "value",
  "type": "rename",
  "optional": true
}
```

#### Write Transformation

```
{
  "scope":
"createEntity",
  "constant": false,
  "targetPath":
"$securityEnabled"
},
{
  "scope":
"createEntity",
  "constant":
"Unified",
  "targetPath":
"$groupTypes[0]"
}
]
```

#### Custom Configurations



Goal	Action	Result
<p>You want Identity Provisioning to read the additional user attributes specified in property <code>aad.user.attributes</code> and write them successfully in the external back-end system.</p>	<p>In the <a href="#">Read Transformation</a>, extend the <code>"user"</code> mapping as follows:</p> <pre> {   "user": {     "condition": "\$userPrincipalName EMPTY false",     "mappings": [       {         "sourcePath": "\$",         "targetPath": "\$"       },       {         "sourcePath": "\$.id",         "targetVariable": "entityIdSourceSystem"       },       ...     ]   } } </pre>	<p>For example, you specify the <code>aad.user.attributes</code> property and set its value to:  <a href="#">id,mail,userPrincipalName,city,department,companyName</a></p> <p>As a result, every user in the external back-end system will have the following attributes populated – ID, e-mail, user principle name, city, department, and company name.</p> <p>Returned information of an exemplary user:</p> <pre> ... {   "Resources": [     {       "id": "555-aaaa-333- abcd-111222333",       "mail": "john.smith@doma in.com",       "userPrincipalName": "ab c@something.onmicrosoft .com",       "city": "Sofia",       "department": "029",       "companyName": "SAP"     }   ] } </pre>

Goal	Action	Result
<p>You want Identity Provisioning to read the additional group attributes specified in property <code>aad.group.attributes</code> and write them successfully in the target system.</p>	<p>Extend the <i>group</i> mapping as follows:</p> <pre> {   "group": {     "ignore": false,     "mappings": [       {         "sourcePath": "\$",         "targetPath": "\$"       },       {         "constant": "urn:ietf:params:scim:schemas:core:2.0:Group",         "targetPath": "\$.\$schemas[0]"       },       ...     ]   } }</pre>	<p>Example: Specify the <code>aad.group.attributes</code> property and set its value to: <i>id,displayName,recommendation,isSubscribedByMail</i></p> <p>As a result, every group in the target system will have the following attributes populated – ID, display name, date and time of the last renewal, and information if it's subscribed by e-mail or not.</p> <p>Returned information of an exemplary group:</p> <pre> ... {   "Resources": [     {       "id": "12345-ccc-000-xyz-777888999",       "displayName": "ImportantGroup3",       "renewedDateTime": "2018-01-01T00:00:00Z",       "isSubscribedByMail": "true"     }   ] }</pre>

Goal	Action	Result
You want the returned value of a group member to be not the ID but a different attribute.	<p>Replace <b>id</b> with the new attribute. For example:</p> <p>If you replace <i>id</i> with <i>mail</i>, the transformation will look like this:</p> <pre>... {   "sourcePath":     "\$.members",   "preserveArrayWithSingleElement": true,   "optional":     true,   "targetPath":     "\$.members"   }, {   "targetPath":     "\$.members[*].mail",   "constant":     "value",   "type":     "rename",   "optional": true } ...</pre>	<p>Returned information of an exemplary group member:</p> <pre>... {   "members": [     {       "id": "5555555-aaaa-333- abcd-1111122223333",       "type": "user",       "value": "johnsmith@mail .acme.com"     }   ] } ...</pre>
	<p><b>Caution</b></p> <p>Make sure that you've added this attribute as a value of property <a href="#">aad.group.member.attributes</a>.</p>	

7. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

SAP Cloud Identity Infrastructure	Neo Environment
<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission <a href="#">Access Proxy System API</a>.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>	<ul style="list-style-type: none"> <li>For AUTH_USER and AUTH_PASSWORD, enter your client ID and secret.</li> <li>For the SCIM_ASSIGNMENT_METHOD constant, make sure the value is <b>PUT</b>.</li> </ul>

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.6.3.40 SCIM System

Follow this procedure to set up SCIM as a proxy system.

### Prerequisites

#### i Note

This system is available for all standalone tenants. Bundle tenants running on SAP Cloud Identity Services infrastructure and Neo environment can use it as a proxy system for reading entities only.

- You have technical user credentials for a SCIM system with read and write access permissions.
- (Optional) You have installed the Cloud Connector in your corporate environment and have done the initial configuration. You need this only if the SCIM system is exposed in a private corporate network. For more information, see [Cloud Connector](#).


#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

Create a general SCIM 2.0 based proxy connector to execute hybrid scenarios. That means, it can provision its entities to another (external) back-end system by request, and then can continue executing CRUD operations back to the SCIM system, whenever the external back-end requests such. This scenario supports provisioning **users** and **groups**.

#### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code [200](#) (OK) with '[totalResults](#)' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code [200](#) (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code [400](#) (Bad Request) with detail error type '[tooMany](#)'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used 'eq' filter).
- Fully qualified names (<schema>:<attribute>) are not supported. For example:  
`GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employee Number eq '<attribute>'`
- If your system supports multivalued e-mails (that is `$.emails[0].value`, `$.emails[1].value`, etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail (`$.emails[0].value`).

### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `scim.user.filter = timezone eq "Africa"`

Then if, for example, the SCIM Proxy endpoint request is: `GET .../Users?filter=userName eq "johnsmith03"`

The query request to the SCIM system API will result into: `/Users?filter=timezone eq "Africa" and userName eq "johnsmith03"`

## Procedure

1. (Optional) Open Cloud Connector to add an access control system mapping for the **SCIM system**. This is needed to allow the Identity Provisioning service to access the SCIM system as a back-end system on the intranet. To learn how, see: [Configure Access Control \(HTTP\)](#)
2. Open your subaccount in SAP BTP cockpit.

If you have a bundle tenant, in the cockpit **Neo > Overview**, you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with "SAP\_BUNDLE\_".

3. Create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

## SAP Cloud Identity Infrastructure

## Neo Environment

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*

For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP BTP, Neo Environment*

For **Basic authentication**, proceed as follows:

1. In SAP Cloud Identity Services admin console, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.
2. Add an administrator user of type **System** and configure the basic authentication method for this user. If you already have a technical user, skip this step.
3. Save your changes.
4. Select your administrator user of type **System** and enable the [Access Proxy System API](#) permission.
5. Save your changes.

For **OAuth authentication**, proceed as follows:

1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ► and choose [Register New Client](#).
2. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
3. From the [Authorization Grant](#) combo box, select **Client Credentials**.
4. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in the external system.
5. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
6. From the left-side navigation, choose ► [Subscriptions](#) ► [Java Applications](#) ► [ipsproxy](#) ►.
7. From the left-side navigation, choose ► [Roles](#) ► [IPS\\_PROXY\\_USER](#) ►.
8. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**. For **<client\_ID>**, enter the one you have saved in the previous main step.

4. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
5. Add [SCIM System](#) as a proxy system. For more information, see [Add a System \[page 1477\]](#).
6. Choose the [Properties](#) tab to configure the connection settings for your system.

### Note

If your tenant is running on SAP BTP, Neo environment, you can create a [connectivity destination](#) in your subaccount in the SAP BTP cockpit, and then select it from the [Destination Name](#) combo box in your Identity Provisioning User Interface.

If one and the same property exists both in the cockpit and in the [Properties](#) tab, the value set in the [Properties](#) tab is considered with higher priority.

We recommend that you use the [Properties](#) tab. Use a connectivity destination only if you need to reuse one and the same configuration for multiple provisioning systems.

## Mandatory Properties

Property Name	Value
Type	Enter: <i>HTTP</i>
URL	Specify the service URL. For example:  <i>http://&lt;cloudfoundry_server&gt;.com/api/uaa/</i>
ProxyType	Depending on your network exposure, enter one of the following: <ul style="list-style-type: none"> <li><i>Internet</i></li> <li><i>OnPremise</i></li> </ul>
Authentication	Enter: <i>BasicAuthentication</i>
User	You can specify one of the following: <ul style="list-style-type: none"> <li>Technical user ID</li> <li>Client ID for OAuth HTTP destinations. It's used for retrieving of the access token.</li> </ul>
Password	(Credential) You can enter one of the following: <ul style="list-style-type: none"> <li>Technical user password</li> <li>Client secret for OAuth HTTP destinations. It's used for retrieving of the access token.</li> </ul>
OAuth2TokenServiceURL	If you need to make OAuth authentication to the system, enter the URL to the access token provider service for OAuth HTTP destinations.  For example: <i>https://&lt;token_provider&gt;.com/api/oauth2/v2.0/token</i>

To learn what additional properties are relevant to this system, see [List of Properties \[page 94\]](#). You can use the main search, or filter properties by the *Name* or *System Type* columns.

## 7. Configure the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The Identity Provisioning offers a default transformation for the *SCIM* proxy system, whose settings are displayed under the *Transformations* tab after saving its initial configuration.

You can change the default transformation mapping rules to reflect your current setup of entities in your SCIM system. For more information, see [Manage Transformations \[page 1494\]](#).

Default read and write transformation:

### → Tip

The proxy *Read Transformation* is used when the external client application (for example, SAP Identity Management) makes initial load. That is, executing GET requests to the resource endpoints (*/Users*



or **/Groups**) to retrieve the corresponding entities of the particular type. The external client application can also execute GET requests to a single resource endpoint (querying a single resource is supported). In this case, the proxy system acts as a *source* one.

The proxy *Write Transformation* is used when the external application manages the entities in the proxy system – creates new entities, updates existing ones, or deletes existing ones. In this case, the proxy system acts as a *target* one.

However, after a *Create* or *Update* operation is performed on the proxy system, the *Read Transformation* is applied to the result, so that the created or updated entity is sent back to the external application. This behavior demonstrates that the proxy *Read Transformation* is used for *write* cases, as well.

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "targetPath": "$",
        "sourcePath": "$"
      },
      {
        "sourcePath":
        "$.id",
        "targetVariable":
        "entityIdSourceSystem"
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [
          {
            "type":
            "concatString",
            "suffix":
            "${entityIdSourceSystem}"
          }
        ],
        "sourcePath":
        "$.userName",
        "targetPath":
        "$.userName",
        "correlationAttribute": true
      },
      {
        "sourcePath":
        "$.emails[0].value",
        "targetPath":
        "$.emails[0].value",
        "optional": true
      },
      {
        "sourcePath":
        "$.emails[?(@.primary==
true)].value",
        "optional": true,
        "correlationAttribute": true
      }
    ],
    "group": {
      "scimEntityEndpoint":
      "Groups",

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint":
    "Users",
    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "sourceVariable":
        "entityIdTargetSystem",
        "targetPath":
        "$.id"
      },
      {
        "condition":
        "$.emails[0].length() > 0",
        "targetPath":
        "$.emails[0].primary",
        "constant": true
      }
    ],
    "group": {
      "scimEntityEndpoint":
      "Groups",
      "mappings": [
        {
          "sourcePath": "$",
          "targetPath": "$"
        },
        {
          "sourceVariable":
          "entityIdTargetSystem",
          "targetPath":
          "$.id"
        }
      ]
    }
  }
}

```

```

    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "sourcePath":
          "$.id",
        "targetVariable":
          "entityIdSourceSystem"
      },
      {
        "sourceVariable":
          "entityBaseLocation",
        "targetVariable":
          "entityLocationSourceSystem",
        "targetPath":
          "$.meta.location",
        "functions": [
          {
            "type":
              "concatString",
            "suffix":
              "${entityIdSourceSystem}"
          }
        ]
      }
    ]
  }
}

```

8. Connect the external consumer to Identity Provisioning with the technical user you have created in step 2.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

#### SAP Cloud Identity Infrastructure

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### Neo Environment

- For AUTH\_USER and AUTH\_PASSWORD, enter your client ID and secret.
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

#### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

### ⚠ Caution

Effective **September 2020**, Shanghai (China) tenants that reside on SAP BTP, Neo environment can be only accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

So make sure you use the correct domain when you construct your REST API requests.

For example: **GET** <https://ipsproxyabcd12345-xyz789.dispatcher.cn1.platform.sapcloud.cn/ipsproxy/api/v1/scim/bbb111aa-1234-aaaa-7777-1234567abcde/Users/s123456789>

To learn more, see: [Proxy Systems \[page 981\]](#)

## Related Information

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## 1.7 Operations

Learn how you, as an administrator, can set up the Identity Provisioning service so that entities from a source system are easily transferred to a target system.

Before triggering provisioning, make sure that you have performed the required setup. For more information, see [Initial Setup of Bundle Tenants \[page 406\]](#).

You can perform the following operations:

- Add source, target and proxy systems.
- Set up configuration properties specific for your systems and scenarios.
- Define mapping rules between the data models of sources and targets.
- Provision entities between systems.
- Manage Deleted Entities
- Configure the frequency of the provisioning processes.

- Run and schedule provisioning jobs.
- View, maintain and delete job logs.
- Handle failed operations
- Enable and disable systems
- Export and import systems
- Update connectors version
- Manage authorizations and certificates
- Migrate a bundle tenant
- Reset the Identity Provisioning UI configurations.
- Reset the Identity Provisioning system.
- Deactivate the Identity Provisioning service.

## Related Information

[System Types \[page 86\]](#)

[Properties \[page 90\]](#)

[Transformations \[page 323\]](#)

### 1.7.1 Add a System

You can add source, target, and proxy systems to the Identity Provisioning UI.

## Context

To provision entities (users, groups, roles) from one system to another across your enterprise, you first need to add and configure these systems as source and target connectors in the Identity Provisioning user interface.

#### ! Restriction

The maximum number of systems you are allowed to add is:

- **20** sources systems
- **50** target systems


If your business requires using more systems, create an incident for component [BC-IAM-IPS](#) to request them. Describe your scenarios and provide a reason why you need the additional systems.

When you add a system, it is created with its default properties and transformations. If the system has different versions (based on the APIs it provides), you can specify which one you want to use, so that the system is created with the version specific properties and transformations.

Versioning is supported for Identity Authentication, SAP SuccessFactors, SAP Concur, SAP Analytics Cloud and SAP Sales Cloud and SAP Service Cloud. It is controlled by the `<system_prefix>.api.version` property.

To add a system, proceed as follows:

## Procedure

1. Access the Identity Provisioning UI and choose a tile – *Source Systems*, *Target Systems*, or *Proxy Systems*.
2. Choose the  *Add* button at the bottom of the left-hand panel.
3. On the *Details* tab, provide the following information:

Field	Description
<i>Type</i>	Select the system type that you want to configure.
<i>System Name</i>	Add a name for your system. Make sure it does not duplicate another system's name in the UI.

**i Note**

System names can have a length of up to 100 characters. Only the following characters are allowed: (a-z), (A-Z), (0-9), (-), (\_), (.) and spaces.

Field	Description
<i>Destination Name</i>	<p>(Optional) Select a destination.</p> <p>If you have previously created a connectivity destination in SAP BTP cockpit on subaccount level, you can access it from the Identity Provisioning UI.</p> <div> <p>→ Remember</p> <ul style="list-style-type: none"> <li>When you select a connectivity destination, it must be compliant to the relevant system type.</li> <li>The destination should specify all the connection settings required for your identity provisioning scenario.</li> <li>For <i>SAP Application Server ABAP</i> systems, creating a destination is <b>mandatory</b>.</li> </ul> </div> <p>If you skip the <i>Destination Name</i> field, you can enter the connection and configuration properties, needed for your scenario, on the <i>Properties</i> tab.</p> <div> <p>i Note</p> <p>If you use both a connectivity destination and the <i>Properties</i> tab, and one and the same property exists in both places, the value set in the <i>Properties</i> tab will be considered with higher priority.</p> <p>If you leave both the <i>Destination Name</i> field and the <i>Properties</i> tab empty and then run a job, no identity provisioning will be performed.</p> </div>
<i>Description</i>	<p>(Optional) Enter a description. It will help you easily distinguish your systems in the list later on.</p>
<i>Source Systems</i>	<p>This field is displayed only for target systems.</p> <p>Select a source system whose identities you want to read and provision to the target one. You can select multiple source systems.</p> <div> <p>→ Tip</p> <p>If you had previously added one or more source systems but some of them were later deleted in your Identity Provisioning UI, an error message will appear. To correct this inconsistency, edit the target system configuration (select active source systems), and save the changes.</p> </div>

- Choose *Save*, if the system you add has no version, or it has two or more versions and you want the default one. The new system appears in the left-side panel. The default transformations and properties are displayed under the respective tabs.

Do not choose [Save](#), if the system you add has two or more versions and you want to specify a particular one. In this case, proceed as follows:

1. From the [Details](#) tab (without saving your configurations), move to the [Properties](#) tab and select it.
2. Add the API version property for your system and a value. For example, if you add SAP SuccessFactors on the [Details](#) tab, add `sf.api.version` and the desired version [2](#).
3. Now, choose [Save](#).

This creates an SAP SuccessFactors system with specific properties and transformation for version 2, which is based on SAP SuccessFactors Workforce SCIM API. Providing value 1, would result in creating a system with specific properties and transformation for version 1, which is based on SAP SuccessFactors HCM Suite OData API.

#### **i Note**

Once you save your configuration, switching between versions is possible but requires manual work, mostly adding the version specific properties and transformations. For more information, see [Update Connector Version \[page 1484\]](#).

5. To add connection and configuration properties, choose ► [Properties](#) ► [Edit](#) ►. See [List of Properties \[page 94\]](#)
6. To modify your default system transformation (if needed), choose ► [Transformations](#) ► [Edit](#) ►.
7. Save your changes.

At the end of the Identity Provisioning URL, a dash-separated string appears. This is the automatically generated unique ID of the newly created system.

## **Related Information**

[Supported Systems \[page 452\]](#)

[Manage Properties \[page 1505\]](#)

[Manage Transformations \[page 1494\]](#)

## **1.7.2 Search and Edit a System**


Admin users can search and edit source, target, and proxy systems in the Identity Provisioning user interface.

### **Prerequisites**

To use the search field, your Identity Provisioning tenant must run on SAP Cloud Identity infrastructure.



## Procedure

1. Access the Identity Provisioning UI and choose a tile – [Source Systems](#), [Target Systems](#), or [Proxy Systems](#).
2. From the list on the left, either directly select a system, or search for it and select it.
3. Select a tab and edit the configurations.
  - [Details](#)
  - [Transformations](#)
  - [Properties](#)
  - [Certificates](#)
4. Choose the  [Edit](#) button and make your changes.
5. Save your changes.

## Related Information

[Manage Properties \[page 1505\]](#)

[Manage Transformations \[page 1494\]](#)

## 1.7.3 Delete a System

You can delete a source, target, or proxy system in the Identity Provisioning UI.

## Context


### i Note

Before you delete a system, make sure you don't need it anymore. If you think you might need it in future, export it first as a JSON or a CSV file. To learn how, see: [Export and Import Systems \[page 1482\]](#)

To delete a system, proceed as follows:

## Procedure

1. Access the Identity Provisioning UI and choose a tile – [Source Systems](#), [Target Systems](#), or [Proxy Systems](#).
2. From the list on the left, select a system.
3. Choose [Edit](#) from the top of the systems panel.

4. At the bottom of the panel, choose the  *Delete* button.
5. In the dialog box, confirm with *OK*.
6. Save your changes. The system disappears from the panel.



## 1.7.4 Enable and Disable Systems

This topic explains how you can enable and disable source and target systems in the Identity Provisioning UI.

### Context

To use a system for provisioning purposes, its status has to be **Enabled**. When you add a new system, it is enabled by default. If one of your added systems is configured and you currently do not need it, but would like to use it later, you can disable it.

### Procedure

1. Access the Identity Provisioning UI and choose a tile – *Source Systems*, *Target Systems*, or *Proxy Systems*.
2. From the list on the left, select a system.
3. Choose *Edit* from the top of the systems panel.
  - If the system is currently disabled, choose the  *Enable* button and confirm with *OK*.
  - If the system is currently enabled, choose the  *Disable* button and confirm with *OK*.
4. Save your changes.

## 1.7.5 Export and Import Systems

This topic explains how you can export and import source, target and proxy systems in the Identity Provisioning UI.


### Context

If you have added and configured a system, you can export it for further use. The export-import function comes in handy in the following use cases:



- You need to back up your system before updating to a new connector version. See: [Update Connector Version \[page 1484\]](#)
- You need another system of the same type but with slightly different setup, and you don't want to manually enter all data and configuration properties all over again.
- You need to reuse an existing system in the Identity Provisioning UI but for another subaccount.
- You have reached the maximum number of systems you are allowed to add. You need to add one more system, which means you must delete some of the previous ones. However, you don't want to lose their configurations, thus you export these systems.

## Procedure

### Export a System

1. Access the Identity Provisioning UI and choose a tile – [Source Systems](#), [Target Systems](#), or [Proxy Systems](#)
2. From the list on the left, select the system you want to export.
3. Choose the  [Export](#) button.
4. The exported system configuration depends on your scenario. If your system is a [source](#) or a [target](#) one, it will be exported as a JSON file. If it's a [proxy](#) one, you have two options:
  - Select [JSON format](#) – the system configuration will be exported as a [.json](#) file, which you can later import back in the Identity Provisioning UI.
  - Select [CSV format](#) – the system configuration will be exported as a [.csv](#) file, which you can later import in the SAP Identity Management UI as a SCIM repository.
5. Save the file on your local file system.

### Import a System

1. Access the Identity Provisioning UI and choose a tile – [Source Systems](#), [Target Systems](#), or [Proxy Systems](#)
2. Choose the  [Add](#) button.
3. In section [Define from File](#), choose the  [Browse](#) button.
4. Browse and select the file with system configuration you need on your local file system. You can import files with extension [.json](#) as well as files with no extension.
5. The system configuration is displayed in the [Details](#) editor. You can also see the imported transformations and properties of this system in the respective UI tabs.
6. Change the [System Name](#), otherwise an error message will appear warning you that a system with this name already exists.
7. The [Properties](#) tab will prompt you to enter the credentials (like passwords or client secrets). When you export a system, credentials are skipped (not displayed as plain text in the [.json](#) file). Therefore, when you import it, you have to manually enter the passwords/secrets.
8. Save your changes.

The imported system appears in the left-side panel. Its ID is different than the one of the "original" system (you can see it in the URL).

#### ⚠ Caution

You cannot export a target system and import it back as a source, nor the other way around.

## 1.7.6 Update Connector Version

Update a connector version to allow your provisioning system to use a new API.

When an SAP cloud solution or service provides a new API for integrating with Identity Provisioning, you can update your respective connector (provisioning system) to use this API by configuring a version property and replacing its transformations.

For example, SAP Sales Cloud and SAP Service Cloud (formerly known as SAP Cloud for Customer) initially provided two SOAP-based APIs for integrating with Identity Provisioning and later introduced a SCIM-based API. Likewise, Identity Authentication service initially provided a SCIM-based API and later introduced an Identity Directory SCIM API.

Version property set to **1** means that your connector is using the initial API. You can continue using it as-is or update your connector to a new version.

#### i Note

Before updating your connector to a new version, it is always a good practice to export the system for backup purposes. See: [Export and Import Systems \[page 1482\]](#)

To update your connector to use a new API, proceed as follows:

### Procedure

1. Access the Identity Provisioning UI and choose a tile – *Source Systems*, *Target Systems*, or *Proxy Systems*.
2. From the list on the left, select a system.
3. On the *Properties* tab, configure the following:
  1. **Version Property** - Add or update the `<system_prefix>.api.version` property and set its value accordingly.

Connector	Property	Value
Identity Authentication	<code>ias.api.version</code>	<ul style="list-style-type: none"> <li>1 - Identity Authentication SCIM API (in short, SCIM API version 1)</li> </ul> <div> <b>i Note</b>  When the property is not defined - Identity Authentication SCIM API is used. </div> <ul style="list-style-type: none"> <li>2 - Identity Directory SCIM API (in short, SCIM API version 2)</li> </ul>
SAP Concur	<code>concur.api.version</code>	<ul style="list-style-type: none"> <li>1 - SAP Concur API (Version 1)</li> </ul> <div> <b>i Note</b>  When the property is not defined - SAP Concur API is used. </div> <ul style="list-style-type: none"> <li>2 - SAP Concur SCIM API (Version 2)</li> </ul>
SAP Sales Cloud and SAP Service Cloud	<code>c4c.api.version</code>	<ul style="list-style-type: none"> <li>1 - Version 1 (SOAP-based API)</li> </ul> <div> <b>i Note</b>  The SOAP-based API version 1 is deprecated. </div> <ul style="list-style-type: none"> <li>2 - Version 2 (SOAP-based API)</li> <li>3 - Version 3 (SCIM 2.0 based API)</li> </ul>
SAP SuccessFactors	<code>sf.api.version</code>	<ul style="list-style-type: none"> <li>1 - SAP SuccessFactors HCM Suite OData API (Version 1)</li> </ul> <div> <b>i Note</b>  When the property is not defined - SAP SuccessFactors HCM Suite OData API is used. </div> <ul style="list-style-type: none"> <li>2 - SAP SuccessFactors Workforce SCIM API (Version 2)</li> </ul>

Connector	Property	Value
SAP Analytics Cloud	sac.api.version	<ul style="list-style-type: none"> <li>• <b>1</b> - SAP Analytics Cloud SCIM API version 1. This is the default value.</li> <li>• <b>2</b> - SAP Analytics Cloud SCIM API version 2</li> </ul>
LDAP Server	ldap.api.version	<ul style="list-style-type: none"> <li>• <b>1</b> - LDAP Server API version 1 is used. This is the default value.</li> <li>• <b>2</b> - LDAP Server API version 2 is used.</li> </ul>
Microsoft Active Directory	ldap.api.version	<ul style="list-style-type: none"> <li>• <b>1</b> - LDAP API version 1 is used. This is the default value.</li> <li>• <b>2</b> - LDAP API version 2 is used.</li> </ul>

2. **Properties with Version-Specific Values** - Update connector properties which have version-specific values.

Connector	Property	Value
SAP SuccessFactors	URL	<ul style="list-style-type: none"> <li>• Version 1: <code>https://apitest.successfactors.com/odata/v2</code></li> <li>• Version 2: <code>https://apitest.successfactors.com</code></li> </ul>
	sf.user.filter	<ul style="list-style-type: none"> <li>• Version 1: <code>username eq 'cbraun'</code> <code>status eq 'active'</code></li> <li>• Version 2: <code>userName eq "cbraun"</code> <code>active eq "true"</code></li> </ul>
SAP Analytics Cloud	csrf.token.path	<ul style="list-style-type: none"> <li>• Version 1: <code>/api/v1/scim/Users?count=1</code></li> <li>• Version 2: <code>/api/v1/scim2/Users?count=1</code></li> </ul>

- On the [Transformations](#) tab, if you've customized your transformation logic, copy and save it first, and then replace it with the default transformation provided for the respective API version. Use the Identity Provisioning connector documentation as a source of information for the transformation you need.
- [Reset the system \[page 1542\]](#) to clear the operational data. It is assumed that you have already run provisioning jobs to target systems.
  - If you reset a target system, set the `ips.delete.existedbefore.entities` to **true**. This ensures that, if from now on you delete entities in the source system that is connected to your target system,

those entities will be recognized as previously existed entities in the target system and will be deleted there.

- If you reset a source system, set the `ips.delete.existedbefore.entities` to **true** in every target system connected to the given source system. This ensures that, if from now on you delete entities in the source system that is connected to your target system, those entities will be recognized as previously existed entities in the target system and will be deleted there.
6. Adapt your new transformation, that is, apply the customizations from your previous transformation.
  7. Save your changes and run a provisioning job.

## 1.7.7 Manage Authorizations

Manage the authorizations of Identity Provisioning administrators, when your bundle or standalone tenant is running on SAP Cloud Identity Services infrastructure or SAP BTP, Neo environment.

### Related Information

[Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#)

[Manage Authorizations in Neo Environment \[page 1490\]](#)

### 1.7.7.1 Manage Authorizations in SAP Cloud Identity Infrastructure

You can request administrative access for your Identity Provisioning bundle or standalone tenant, add additional users as administrators of the tenant and create a technical user with the necessary authorizations for configuring real-time provisioning and proxy systems.

### Prerequisites

- Ensure your tenant is running on SAP Cloud Identity Services infrastructure. For more information, see [Tenant Model \[page 8\]](#)

#### **i** Note

When the Identity Provisioning tenant is initially provisioned to your organization, only one user is added as a tenant administrator. After that, due to possible legal and security issues, SAP adds additional tenant administrators only in exceptional cases (for example, the existing administrator left the company, or for some reason there is no active administrator for this tenant).

To avoid access-related issues in such cases, it is always a good practice for you to assign more than one administrators. Adding additional ones is exclusively in the responsibility of the current tenant administrators. For more information, see the *Add Additional Admin Users* section below.

## Get Administrative Access

To get administrative access for the Identity Provisioning tenant, proceed as follows:

### Procedure

1. Sign in to SAP Cloud Identity Services administration console and navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.

For more information about your tenants, see [Viewing Assigned Tenants and Administrators](#)

2. Add an administrator of type [User](#) and enable the [Manage Identity Provisioning](#) role.

You are now granted the main IPS\_ADMIN role.

3. Save your changes.

## Add Additional Admin Users

To add additional users as administrators of the Identity Provisioning tenant, proceed as follows:

### Procedure

1. Sign in to SAP Cloud Identity Services administration console and navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.

2. Add the new administrator of type [User](#) and enable the [Manage Identity Provisioning](#) role.

This administrator is now granted the main IPS\_ADMIN role.

3. Save your changes.

## Create a Technical User

To create a technical user with the necessary authorizations for configuring real-time provisioning and proxy systems, proceed as follows:

### Procedure

1. Sign in to SAP Cloud Identity Services administration console and navigate to ► [Users & Authorizations](#) ► [Administrators](#) ►.






2. Add the new administrator of type [System](#).

This is the technical user you can use for configuring provisioning scenarios with proxy systems and real-time provisioning.

For more information, see [Add System as Administrator](#)

3. In the [Configure System Authentication](#) screen, configure certificate-based authentication or basic authentication for the technical user.
4. Assign the necessary authorizations (roles) to the new user.

Role	Description
<a href="#">Access Proxy System API</a>	<p>Authorizations to access API for provisioning identities via proxy systems</p> <p>This role is needed for provisioning scenarios where proxy systems in the Identity Provisioning admin console are configured for synchronizing user data to and from central identity management solutions (such as, the on-premise SAP Identity Management).</p> <p>In this case, you use the credentials of the admin user with <a href="#">Access Proxy System API</a> role assigned for setting up the technical user in the identity management solution for communicating with Identity Provisioning.</p>
<a href="#">Access Real-Time Provisioning API</a>	<p>Authorizations to access API for real-time provisioning of identities</p> <p>This role is needed for provisioning scenarios where user data is provisioned real-time without running jobs (manual or scheduled ones) in Identity Provisioning.</p> <p>In this case, you use the credentials of the admin user with <a href="#">Access Real-Time Provisioning API</a> role assigned for setting up the authentication mechanism of the provisioning system defined on the <a href="#">User Provisioning</a> screen in the Identity Authentication admin console.</p>

Role	Description
<a href="#">Access Identity Provisioning Tenant Admin API</a>	<p>Authorizations to access tenant API for running Identity Provisioning jobs or downloading job logs.</p> <p>This role is needed for the following tasks:</p> <ul style="list-style-type: none"> <li>Running provisioning jobs from an API client</li> <li>Downloading provisioning job logs from an API client</li> </ul> <p>The API is available on the SAP Business Accelerator Hub: <a href="#">SAP Cloud Identity Services</a>  <a href="#">Identity Provisioning Service</a>  <a href="#">API Reference</a> .</p> <ul style="list-style-type: none"> <li>For running jobs, select <a href="#">Jobs</a> and use the following URL pattern: <code>https://&lt;ias-tenant-host&gt;/ips/service/publicapi/v1/startJob/{SourceSystemId}/jobs/{JobType}</code> For more information, see <a href="#">Run Provisioning Jobs via API [page 1532]</a></li> <li>For downloading job logs, select <a href="#">JobLogs</a> and use the following URL pattern: <code>https://&lt;ias-tenant-host&gt;/ips/service/publicapi/v1/jobLogs/&lt;JobId&gt;?action=export&amp;logType=&lt;logType&gt;</code> For more information, see <a href="#">Manage Provisioning Job Logs [page 1600]</a></li> </ul>

5. Save your changes.


## 1.7.7.2 Manage Authorizations in Neo Environment

Manage the authorizations of Identity Provisioning administrators, when your bundle or standalone tenant is running on SAP BTP, Neo environment.

### Prerequisites

- Ensure your tenant is running on SAP BTP, Neo environment. For more information, see [Tenant Model \[page 8\]](#)

#### i Note

Only users authenticated by [SAP ID Service](#)  can be added as Identity Provisioning administrators (for example, S-users).

## Note


When the Identity Provisioning tenant is initially provisioned to your organization, only one user is added as a tenant administrator. After that, due to possible legal and security issues, SAP adds additional tenant administrators only in exceptional cases (for example, the existing administrator left the company, or for some reason there is no active administrator for this tenant).

To avoid access-related issues in such cases, it is always a good practice for you to assign more than one administrators. Adding additional ones is exclusively in the responsibility of the current tenant administrators. For more information, see the procedures for bundle and standalone tenants below.

## Bundle Tenants

In addition to the *Manage Identity Provisioning* role, you can assign yourself other authorizations for managing on-premise connections, OAuth clients and destinations, and provide additional users with administration rights for the Identity Provisioning bundle tenant.

## Procedure

1. Log on to the *Identity Provisioning* admin console and go to ► *Security* ► *Authorizations* ►.
2. Choose  *Add*.
3. In the *User ID* field, enter the ID of a user you want to authorize (for example, **p1234567890**).
4. (Optional) In the *Display Name* field, enter a human readable name.
5. In the *Configure Authorizations* screen, choose *Edit* and assign the necessary authorizations (roles) to the new admin user.

Role	Description
<i>Manage Identity Provisioning</i>	If you set this option to <i>ON</i> , the new admin users will access Identity Provisioning from your subaccount (the URL you use) – they will be allowed to add and configure systems, run provisioning jobs, view job logs, and add other users as administrators for your subaccount.

Role	Description
<a href="#">Manage On-Premise Connections</a>	<p>If you set this option to <a href="#">ON</a>, the new admin users will be able to connect the Cloud Connector to your subaccount. The Cloud Connector is needed for the communication between cloud and on-premise systems.</p> <p>You can see your currently configured Cloud Connector connections in SAP BTP cockpit. Go to your subaccount and choose ► <a href="#">Connectivity</a> ► <a href="#">Cloud Connectors</a> ►.</p> <p>Your subaccount URL follows the pattern: <code>https://account.&lt;region-host&gt;/cockpit#/acc/&lt;ips-tenant-ID&gt;</code></p> <p>For example:  <code>https://account.eu2.hana.ondemand.com/cockpit#/acc/abc123456</code></p> <p>For more information, see: <a href="#">Cloud Connector: Initial Configuration</a></p>
<a href="#">Manage OAuth Clients</a>	<p>If you set this option to <a href="#">ON</a>, the new admin users will be allowed to register OAuth clients. This is needed for:</p> <ul style="list-style-type: none"> <li>• Real-time provisioning</li> <li>• Proxy scenarios</li> </ul> <p>You can see your currently configured OAuth application client credentials in SAP BTP cockpit. Navigate to your subaccount, and choose ► <a href="#">Security</a> ► <a href="#">OAuth</a> ►.</p> <p>Your subaccount URL follows the pattern: <code>https://account.&lt;region-host&gt;/cockpit#/acc/&lt;ips-tenant-ID&gt;</code></p> <p>For example:  <code>https://account.eu2.hana.ondemand.com/cockpit#/acc/abc123456</code></p> <p>To learn how to create an OAuth client, see: <a href="#">Register an OAuth Client</a></p> <div> <p>→ Tip</p> <p>From the <a href="#">Subscription</a> combo box, select <b>&lt;provider_subaccount&gt;/ipsproxy</b>.</p> </div>

Role	Description
<a href="#">Manage Destinations</a>	<p>If you set this option to <a href="#">ON</a>, the new admin users will be able to create connectivity destinations in SAP BTP cockpit. Creating a destination is <b>mandatory</b> for configuring <a href="#">SAP Application Server ABAP</a> provisioning systems and on-premise systems using the Cloud Connector for which a <a href="#">Location ID</a> is configured.</p> <p>Also, use a connectivity destination if you need to reuse one and the same configuration for multiple provisioning systems. In all other cases, we recommend that you use the <a href="#">Properties</a> tab for configuring connection details.</p> <p>To create a connectivity destination, go to the SAP BTP cockpit, navigate to your subaccount, and choose <a href="#">► Connectivity ► Destinations ►</a>.</p> <p>Your subaccount URL follows the pattern: <code>https://account.&lt;region-host&gt;/cockpit#/acc/&lt;ips-tenant-ID&gt;</code></p> <p>For example:  <code>https://account.eu2.hana.ondemand.com/cockpit#/acc/abc123456</code></p> <p>To learn how to create an RFC destination for your AS ABAP system, see: <a href="#">Create RFC Destinations</a></p>

6. Save your changes.
7. Repeat this procedure for every user you want to authorize. You cannot authorize multiple users at once.

## Standalone Tenants

Provide additional users with administration rights for your Identity Provisioning subaccount.

### Procedure

1. Open your subaccount in SAP BTP cockpit.
2. From the left-side navigation, choose [► Applications ► Subscriptions ►](#).
3. Choose your Java application ([ips](#), [ipsproxy](#), or [idds](#)).
4. Choose [Roles](#).
5. Select the relevant role, choose [Assign](#) and add a user ID.

Role	Description
IPS_ADMIN	<p>The main administrator role. It provides you with access to all Identity Provisioning UI systems and features. You can manage source, target and proxy systems, run and schedule jobs, view and maintain job logs, and reset your Identity Provisioning configurations.</p> <p>The role is available if you subscribe to the <i>ips</i> and <i>ipsproxy</i> application.</p>
IPS_PROXY_USER	<p>This role allows you to provision entities from and to proxy systems via proxy system APIs.</p> <p>The role is available if you subscribe to the <i>ipsproxy</i> application.</p>
SCIM_READ	<p>This role allows you to read entities from <i>Local Identity Directory</i> when you set it as a source system.</p> <p>The role is available if you subscribe to the <i>idds</i> application.</p>
SCIM_MANAGE	<p>This role allows you to perform CRUD operations to entities when you call the <i>Local Identity Directory</i> through SCIM API requests.</p> <p>The role is available if you subscribe to the <i>idds</i> application.</p>

6. The new user is now granted the selected role.

## 1.7.8 Manage Transformations

You can manage transformations with graphical and JSON text editor. Regardless of which one you choose, the following initial steps are the same.

1. Access the Identity Provisioning UI and choose a tile – *Source Systems*, *Target Systems*, or *Proxy Systems*
2. Select a system from the left panel and go to the *Transformations* tab.  
The graphical editor is displayed by default. You can switch to the JSON editor by choosing the code-bracket icon.
3. Choose *Edit*. You need to work in edit mode to add, modify and delete entities and their configurations.
  - Working with the JSON editor allows you to type changes and perform operations like select, cut, copy and paste the transformation code.
  - Working with the graphical editor allows you to graphically model your changes.
4. Save your changes.

## Related Information

[Transformation Editors \[page 402\]](#)

[Working with Graphical Editor \[page 1495\]](#)

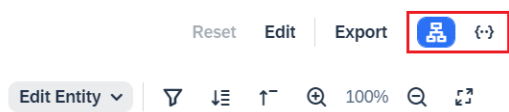
### 1.7.8.1 Working with Graphical Editor

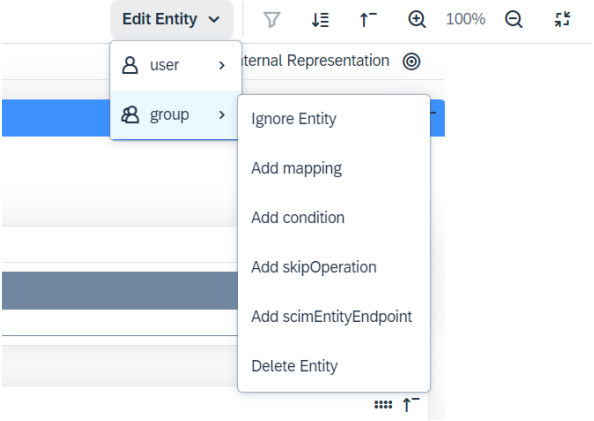
You can create, update and delete entities and their attribute mappings with a handy and easy to use graphical editor. It provides typical operations for an editor, like adding new data, editing and deleting existing data and saving changes. And what's more, it brings improved user experience, requires less typing and more choosing from a list of prefilled values.

#### Basic Operations

##### Note

The graphical editor is available only for Identity Provisioning tenants running on SAP Cloud Identity infrastructure. It is the default editor.

Description	Operation
<b>Switch Editor</b>  When selecting the <i>Transformations</i> tab, the graphical editor is displayed by default. It is represented by a tree icon (in the red rectangle). You can switch to the JSON text editor by choosing the code-bracket icon.  Switching between editors is allowed in view and edit mode.	

Description	Operation
<p>▶▶ <a href="#">Edit Entity</a> ▶ <a href="#">user/group</a> ▶ <a href="#">Ignore Entity</a> ▶</p> <p>Use this option to ignore the provisioning of an entity. When selected, a check mark appears in front of the option and the entity is greyed out. To enable it again, deselect the option.</p>	 <p>The screenshot shows the 'Edit Entity' dropdown menu with the 'group' option selected. The menu includes options: 'Ignore Entity', 'Add mapping', 'Add condition', 'Add skipOperation', 'Add scimEntityEndpoint', and 'Delete Entity'. The 'Ignore Entity' option is highlighted with a checkmark.</p>
<p>▶▶ <a href="#">Edit Entity</a> ▶ <a href="#">user/group</a> ▶ <a href="#">Add mapping</a> ▶</p> <p>Use this option to add attribute mapping for an entity. When selected, the <a href="#">Add Mapping</a> dialog appears. For more information, see <a href="#">Add or Edit Attribute Mappings</a>.</p> <p>New mappings are added at the end of the list of mappings for the selected entity. As the order matters, you can move them up and down to define the proper place for processing them.</p>	
<p>▶▶ <a href="#">Edit Entity</a> ▶ <a href="#">user/group</a> ▶ <a href="#">Add condition</a> ▶</p> <p>Use this option to add a condition for an entity. When selected, the <a href="#">Add Condition</a> dialog appears. You need to provide a value. Conditions defined on an entity level are displayed right under the entity name in orange box.</p>	
<p>▶▶ <a href="#">Edit Entity</a> ▶ <a href="#">user/group</a> ▶ <a href="#">Add skipOperation</a> ▶</p> <p>Use this option to add skipOperation for an entity. It is always defined on an entity level and in target system transformations only. When selected, the <a href="#">Add skipOperation</a> dialog appears with a dropdown list of prefilled values.</p>	
<p>▶▶ <a href="#">Edit Entity</a> ▶ <a href="#">user/group</a> ▶ <a href="#">Add scimEntityEndpoint</a> ▶</p> <p>Use this option to add the entity endpoint which is needed in proxy systems only.</p>	
<p>▶▶ <a href="#">Edit Entity</a> ▶ <a href="#">user/group</a> ▶ <a href="#">Delete Entity</a> ▶</p> <p>Use this option to delete an entire entity. Confirmation is required. When an entity is deleted, you cannot restore it. You can only add a new one.</p>	



## Description

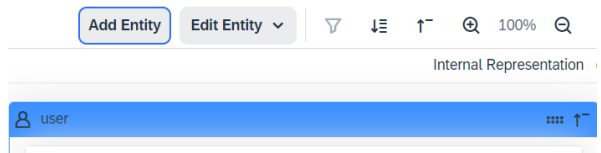
## Operation

### Add Entity

#### i Note

The **Add Entity** button is displayed only if an entity has been deleted.

When selected, an **Add Entity** dialog appears. You are allowed to select only an entity of the type that has been deleted. The new entity appears in its dedicated colored box and has no content.



### Filter

The **Filter** icon allows you to search for entity attributes by their scope: create, patch and delete. Filtering is enabled only in view mode, meaning you must not choose **Edit** on the **Transformations** tab.

When selected, the **Filter** dialog appears with the default value (**Not Filtered**). Selecting a filter results in returning all entity attributes with the given scope plus all attributes without any scope.



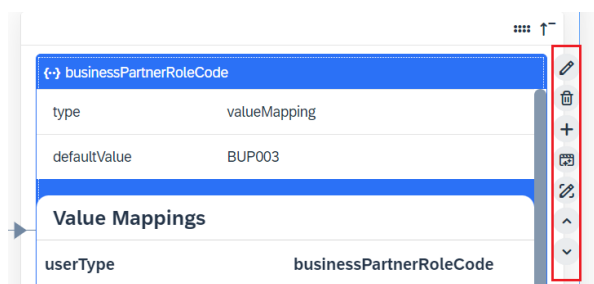
#### → Remember

Attribute mappings with no scope are always returned regardless of the selected filter.

Choosing **Reset** returns you to the default value (**Not Filtered**).

Operations on an attribute mapping level:

- **Pencil** icon - Edit an attribute mapping
- **Trash** icon - Delete an attribute mapping
- **Plus** sign icon - Adds an attribute mapping
- **Function** icon - Add or edit a function  
Functions appear only for attributes on the right side of the editor.
- **Pencil** icon - Edit value mappings  
This icon appears only for attributes with type set to `valueMapping`, for example: `businessPartnerRoleCode`, `timezone` and others.
- **Up arrow** icon - Move an attribute mapping one level up.
- **Down arrow** icon - Move an attribute mapping one level down.



## Add or Edit Attribute Mappings

When you select [Add mapping](#) for an entity or choose the [Plus](#) icon next to an attribute, the [Add Mapping](#) or [Edit Mapping](#) dialog appears, respectively. Use it to configure the following:

- Add [Mapping Paths](#) - A mapping path is a name-value pair (that is, source and target path expressions and their values) based on a selected type of value.  
In the [Name](#) field, you can choose one of the predefined `sourcePath` and `targetPath` expressions, select your [Type](#) of value: [attribute](#), [property](#) or [value](#) and finally, provide the value itself.  
Attribute is the default type value.  
If you want to configure a complex attribute, for example `name.familyName`, add the `name` attribute first and then choose the [Plus](#) sign icon to add the `familyName` subattribute second. You can add as many subattributes as you need.
- Add [Mapping Configurations](#) - A mapping configuration is a name-value pair (that is, a transformation expression and its value).  
In the [Name](#) field, you can choose one of the predefined expressions: [condition](#), [scope](#), [constant](#), [ignore](#), [correlationAttribute](#) and others and provide a value. Some expressions have predefined values. For example, ignore, optional, scope, skipOperations.
- View your configuration in the [Mapping Representation in JSON](#) view.

## Add or Edit Functions

When you choose the [Function](#) icon on an attribute mapping level, the [Add Functions](#) dialog appears. If a function is already configured for a mapping, the [Edit Functions](#) dialog appears. Use it to configure the following:

- Choose [Add](#) [Function](#) and select one from the list.  
Most functions have predefined parameters, listed in the [Name](#) field. For example, if you add the `replaceLastString` function, the `regex` and `replacement` parameters are prefilled. You only need to provide a value.  
If no parameters are listed in the [Name](#) field, choose [Add](#) next to the selected function to add a name-value pair.  
You can add as many functions as you need. The number of functions you've added is displayed under the title of the dialog.
- Order the functions if needed. As the order of the functions matter, you can choose the arrows to move them up and down to define the proper place for processing them.

## Related Information

[Blog Post: Identity Provisioning Graphical Editor – the Game Changer for Transformation Configurations](#) 

## 1.7.8.2 Enabling Group Assignment

When provisioning users from source to target systems, in addition to replicating the user data, you can also assign them to groups in the target system. This works for standard provisioning with or without configured bulk operation on the target, and real-time provisioning.

### Prerequisites

- You have configured a SCIM-based target system which supports patch operation.

#### i Note

The following is a list of the supported SCIM-based target systems: SAP Ariba Applications, SAP Build Work Zone, standard edition, SAP Commerce Cloud, SAP Field Service Management, SCIM System, Identity Authentication (using SCIM API version 2), SAP SuccessFactors Learning, SAP SuccessFactors, SAP Analytics Cloud, SAP BTP XS Advanced UAA (Cloud Foundry) and Cloud Foundry UAA Server.

- The groups that you want to assign exist on the target system.

### Context

To enable the group assignment, you need to modify the target system transformation by adding a mapping under the user resource containing the following configuration parts:

- **Condition** - Defines for which users the condition will apply. For example, all users with emails.
- **Constant** - Holds the IDs of the groups in the target system. Currently, you can only identify groups by ID.
- **targetVariable** - Specifies whether you want to assign users to groups or unassign users from groups.

## Assign Users to Groups

This mapping will result in assigning all users with emails to two groups in a target system.

### Code Syntax

```
{
  "user": {
    "mappings": [
      <Existing transformation>
      {
        "condition": "($.emails
EMPTY false)",
        "constant": [
          {
            "id": "00f8ab94-
a732-48fa-9169-e51f87b8dcd5"
          },
          {
            "id": "01231139-4711-4a28-8f9d-67458
43ef716"
          }
        ],
        "targetVariable": "assignGroup"
      }
    ],
    "group": {
      "mappings": [
```

## Unassign Users from Groups

This mapping will result in unassigning all users with emails from two groups in a target system.

### Code Syntax

```
{
  "user": {
    "mappings": [
      <Existing transformation>
      {
        "condition": "($.emails
EMPTY false)",
        "constant": [
          {
            "id": "006aeaea-3486-479e-8c65-
eabd2868653e"
          },
          {
            "id": "00bbae67-6b9d-4f46-9306-6aad2
8c40861"
          }
        ],
        "targetVariable": "unassignGroup"
      }
    ],
    "group": {
      "mappings": [
```

Groups which are assigned to users through the user resource are referred to as user managed groups. Following a successful provisioning, if the user managed groups are managed through the group resource, the assignments made through the user managed groups will be overwritten.

### Note

- If the group doesn't exist in the target system during user creation or update, the user provisioning fails.
- If the group doesn't exist in the target system during user deletion, the user deletion succeeds.

If you want to remove the group assignments of a deleted user from the source system, you need to modify the target system transformation by adding a mapping under the user resource containing the following configuration parts:

- Constant** - Holds the IDs of the groups in the target system which will be updated. Currently, you can only identify groups by ID.
- targetVariable** - Use **"unassignGroup"** to specify that you want to unassign users from the defined groups.
- scope** - Use **"deleteEntity"** to define that the deleted user from the source system should be persisted in the target system.

```
{
  "constant": [
```

```

    {
      "id": "groupID"
    }
  ],
  "targetVariable": "unassignGroup",
  "scope": "deleteEntity"
}

```

This mapping will result in updating the assignments to the defined group in the target system.

When using the group unassignment for users that are deleted from the source system, these are the possible scenarios based on the system where the user was created and the value of the property `ips.delete.existedbefore.entities`:

- When the user was provisioned from another system by the Identity Provisioning service or it was initially created in the target system and the property `ips.delete.existedbefore.entities` is set to **true**. In this case only transformation mappings for the affected user containing scope `deleteEntity` are executed. This mapping could be used, for example, for disabling a user account in the target system. For more information, see [Transformation Expressions \[page 330\]](#) → `deleteEntity`.

#### i Note


The update is executed via PUT or PATCH operation, based on the configuration of the property `*.support.patch.operation` in the target system.

- When the user was initially created in the target system and the property `ips.delete.existedbefore.entities` is set to **false**, the group assignments of the user in the target system will be updated.

## Related Information

[Manage Transformations \[page 1494\]](#)

[Transformation Expressions \[page 330\]](#)

**Blog Post:** [Group Assignments Based on User Attributes – a Flexible Solution for Managing Conditional & Risk-Based Authentication and Much More](#) 

### 1.7.8.3 Manage Transformations History

Manage the history of transformations, review and restore them to a previous version of your choice.

#### i Note

This functionality is available for Identity Provisioning tenants running on SAP Cloud Identity infrastructure. Customers with tenants running on SAP BTP, Neo environment can only reset modified transformations to their initial state. For more information, see [Reset Identity Provisioning Transformations \[page 1504\]](#)

## Context

Every time you modify the transformation of a provisioning system and save your changes, a new version is created and displayed in the [Transformation History](#) screen. It lists all versions - from the initial to the current one, and gives you additional information about the last modification date, the person who modified it and a description. Within this screen, you can perform a number of actions: reset the transformation to its initial version, apply the current default transformation or apply a version of your choice, as well as download previous versions of the transformations for the current system. In case you need to compare two or more versions, you can do it in a text editor, once you download the respective versions.

### Transformation History

Column	Description
<a href="#">Version</a>	<p>Transformation version</p> <div><p><b>i Note</b></p><p>Version 1 is the initial version. The version with the highest number is the current version.</p></div> <p>The <b>initial version</b> is the one that you get when the system is created (manually or automatically). If you don't make any changes to the transformation while creating the system, the initial transformation matches the default one. If you or a provisioning tool make changes to the transformation while creating the system, the initial transformation doesn't match the default one.</p> <p>There is no restriction on the number of versions you can store and manage. By default, 20 versions are displayed on the screen. You need to expand the list until all your versions are displayed.</p>
<a href="#">Last Modified</a>	Date and time the transformation was last modified
<a href="#">Modified By</a>	<p>Who modified the transformation:</p> <ul style="list-style-type: none"><li>• The userID of the person who modified the transformation</li><li>• <b>SAP</b> - Indicates that this transformation has been migrated by SAP. This could be your initial transformation or your current one.</li></ul>
<a href="#">Description</a>	<p>(Mandatory) Description of the changes</p> <p>When modifying the transformation, saving your changes is only allowed if you provide a description.</p>

Column	Description
<a href="#">Actions</a>	<p>Actions you can perform for a selected version:</p> <ul style="list-style-type: none"> <li>• Apply the selected version. This restores the transformation to the selected version.</li> <li>• Download the selected version.</li> </ul>
<a href="#">Default Version</a>	Apply the current default version.
<a href="#">Reset to Initial</a>	Reset the transformation to its initial version.

Following the introduction of transformations history on November 27, 2023, all your systems will have initial version listed in the [Transformation History](#). What this initial version means for you, however, depends on the release of the reset transformation functionality on November 1, 2021. And that is because with implementing the reset, Identity Provisioning began storing the initial version of modified transformations.

- Systems created before November 1, 2021, will get the current transformation as initial version.
- Systems created after November 1, 2021, will get their actual initial transformation (the one they got at system creation) as initial version. If this transformation has been changed, the [Transformation History](#) will also list the current transformation as version 2.
- As of November 27, 2023, every newly created system is created with initial version. Disabled systems will get the initial version upon enabling them.

### **i** Note

Deleting a system or resetting the tenant results in deleting the transformation history. Resetting the system itself does not delete the history.

## Procedure

To manage your transformations history and apply the version you need, proceed as follows:

1. Sign in to SAP Cloud Identity Services administration console and navigate to [Identity Provisioning](#).
2. Select a system under [Source Systems](#), [Target Systems](#) or [Proxy Systems](#) and choose the [Transformations](#) tab.  
If the transformations have been modified, an info message is displayed in the message view.
3. Select the [Edit](#) button and choose the clock icon.  
This opens the [Transformation History](#) screen.  
Without entering the edit mode, you are only allowed to view the transformation versions and download them.
4. Choose the action that you want to execute.
5. Choose [Close](#).

## 1.7.8.4 Reset Identity Provisioning Transformations

Resetting Identity Provisioning system transformations restores them to their initial state.

### Context

An initial state is defined as follows:

- The default transformation when the system was created in the Identity Provisioning UI.
- The transformation when the system was automatically created in an Identity Provisioning bundle tenant. This could be either the default transformation for the given system or a transformation provided specifically for it.

Resetting transformations is available only for modified transformations in newly created source, target and proxy systems after November 1, 2021. Before doing that, it is always a good practice to copy and save your transformations (or export the system), in case you need to get back to them later. Once the reset is complete, your modified transformations will be deleted.

Proceed as follows:

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Access the Identity Provisioning UI and choose a tile – *Source Systems*, *Target Systems*, or *Proxy Systems*.
3. Select a system and choose the *Transformations* tab.

If the transformations have been modified, a message informs you about that, and the *Reset* button is displayed on the right of the screen. The button is grayed out.

4. Select the *Edit* button.

This is needed because resetting transformations requires that you start working in edit mode.

5. Select the *Reset* button and confirm your choice in the *Approve* dialog.
6. Save your changes.

### Next Steps

After the reset is complete, you can start fresh with your transformations. You can modify them again to meet your provisioning scenarios needs and run a provisioning job.

Note that starting fresh with your transformations doesn't mean starting fresh with your system. Even though you reset the transformations of a given system and run a provisioning job, Identity Provisioning still keeps the



operation data for that system (for example, which entities have been provisioned and whether they exist, and therefore need to be updated or deleted).

If you want to delete the operation data and start fresh with your system, you need to reset the system. For more information, see: [Reset Identity Provisioning System \[page 1542\]](#)


## 1.7.9 Manage Properties

You can add, delete and modify properties for a system in the Identity Provisioning UI.

### Prerequisites

You have added a system (source, target, or proxy) in the Identity Provisioning user interface. To learn how, see [Add a System \[page 1477\]](#).

### Procedure

1. Access the Identity Provisioning UI and choose a tile – [Source Systems](#), [Target Systems](#), or [Proxy Systems](#)
2. Select a system from the left panel and go to the [Properties](#) tab.
3. To modify the current properties, choose  [Edit](#) in the bottom right corner.

#### i Note

When you update the URL or the host name of an existing provisioning system, you must re-enter the values of the configured credential properties. The only exception to this are the credential properties of systems that are created with a connectivity destination.

4. Add the properties required by your scenario to make successful connection to the selected system. You can use two types of properties:
  - [Standard](#): These are properties, whose values are displayed as numbers or plain text strings. For example: **Type**, **ProxyType**, **URL**, **Authentication**.
  - [Credential](#): These are properties whose values contain sensitive information that must not be displayed as plain text. For example: **Password** (standard passwords, private keys, or OAuth client secrets), **ssh.private.key** (relevant to SSH Server), **hana.jdbc.ssh.tunnel.private.key** (relevant to SAP HANA Database).

#### i Note

Properties whose values contain sensitive information can be added only as [Credential](#) in the Identity Provisioning UI. The values of these properties are stored as encrypted data. They are excluded from the file during system configuration export.

5. Save your changes.

## Related Information

[List of Properties \[page 94\]](#)

[Supported Systems \[page 452\]](#)

## 1.7.10 Manage Certificates

Identity Provisioning supports certificate-based authentication for secure communication with the provisioning systems (connectors) provided by the service.

### Context

Certificates can be used in outbound and inbound connections to Identity Provisioning.

In **outbound connections**, Identity Provisioning acts as an SSL client. The service generates an X.509 client certificate for mutual Transport Layer Security (mTLS) authentication against a given provisioning system acting as a server. The Identity Provisioning client certificate must be uploaded to the given provisioning system for configuring the certificate-based authentication there. For example, in SAP BTP ABAP Environment, the Identity Provisioning certificate must be uploaded to the communication user used in the communication arrangement.

In **inbound connections**, Identity Provisioning acts as a server whereas the given provisioning system acts as a client and must present a client certificate for establishing the communication to the service. Customers of bundle and standalone tenants running on SAP BTP, Neo environment import client certificates in the Identity Provisioning admin console. Customers of bundle and standalone tenants running on SAP Cloud Identity infrastructure upload the client certificates in the Identity Authentication admin console on the technical user of type [System](#).

Inbound certificates are supported for source and proxy systems in the following scenarios: configuring proxy systems and real-time provisioning.

#### i Note

Client certificate authentication is not supported for systems where the [ProxyType](#) and [ldap.proxyType](#) properties, required for the HTTP and LDAP connection respectively, are set to **OnPremise**.

#### i Note

If you are using Identity Provisioning proxy and real-time provisioning scenarios, ensure that you trust the new root CA: DigiCert Global Root G2 as the old one DigiCert Global Root CA will be deprecated. For more information, see [Root Certificate Replacement](#).

## Related Information

[Generate and Manage Certificates for Outbound Connection \[page 1507\]](#)

[Manage Certificates for Inbound Connection \[page 1510\]](#)

### 1.7.10.1 Generate and Manage Certificates for Outbound Connection

Identity Provisioning handles the following tasks related to X.509 client certificates for outbound connection: generating, automatic regenerating, activating, deactivating, downloading and deleting.

Administrators can manage up to two certificates for secure outbound connection with a given provisioning system. Only one certificate can be active at a time.

Generating a second certificate might be needed in case your active certificate is about to expire. Currently, renewing or extending the validity of a certificate is handled by generating a second one. This will ensure that the communication between Identity Provisioning and the given provisioning system won't be disrupted.

Both certificates - the first and the second one, have the same distinguished name (DN), issued by the same certificate authority (CA).

Four weeks before the expiration of your active certificate you'll start receiving a warning message in the [Certificates](#) tab of your Identity Provisioning user interface (UI). If you allow the certificate to expire, it becomes invalid. It is not possible to extend its validity.

#### → Recommendation

Enable [Automatic Regeneration](#) to ensure your certificate does not expire. This option is supported for tenants running on SAP Cloud Identity Services infrastructure.

The following table explains the key tasks available:

Task	Description
<a href="#">Automatic Regeneration</a>	<p>Turn automatic regeneration option on or off:</p> <ul style="list-style-type: none"> <li>• <b>ON</b> - enables the automatic regeneration and activation of outbound certificates. The certificate will be automatically regenerated and activated within 14 days prior to its expiration.</li> </ul> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p>→ <b>Recommendation</b></p> <p>This option is recommended for systems validating certificates based on their subject and issuer. In such cases, no further manual steps are required.</p> <p>This option is not recommended for systems validating certificates based on fingerprint. When you regenerate the certificate, the fingerprint is changed. This will cause the authentication to fail until the certificate is downloaded and uploaded to the connected system.</p> </div> <p>If your active certificate has expired and you have manually generated a second one without activating it, enabling the automatic regeneration option will result in the following behavior:</p> <ol style="list-style-type: none"> <li>1. A new certificate will be generated and set to active.</li> <li>2. The expired certificate will be set to inactive.</li> <li>3. The initially inactive certificate will be deleted.</li> </ol> <ul style="list-style-type: none"> <li>• <b>OFF</b> - disables the automatic regeneration. Certificate regeneration will not be done automatically, therefore checking certificate expiration and regeneration should be a manual process.</li> </ul>
<a href="#">Generate</a>	<p>You can generate a maximum of two certificates. After reaching the limit, the <a href="#">Generate</a> button is greyed out.</p> <p>The status of the first certificate you generate is set to <a href="#">Active</a>. The status of the second certificate is set to <a href="#">Inactive</a>.</p> <p>Certificates are distinguished by their <b>fingerprints</b> (the unique identifier of the certificate).</p>

Task	Description
<a href="#">Activate</a>	<p>Activate an inactive certificate.</p> <div> <p>→ Recommendation</p> <p>We recommend that you activate your certificate after you have uploaded it in the backend provisioning system. This will ensure that the communication between Identity Provisioning and the backend system won't be disrupted.</p> </div> <p>Only one certificate can be active at a time.</p>
<a href="#">Download</a>	You can download both active and inactive certificates.
<a href="#">Expand</a>	<p>You can expand active and inactive certificates to view their details.</p> <p>If a certificate is about to expire, you see a warning message in the details.</p>
<a href="#">Delete</a>	You can delete only inactive certificates.

## Generate Certificate

To generate a client certificate, proceed as follows:

1. In the Identity Provisioning UI, select the provisioning system that you want to configure client certificate authentication for (SAP BTP ABAP Environment, for example).
2. Select the [Certificate](#) tab and choose [Generate](#).  
If the certificate is generated successfully, the toast message `Certificate generated successfully` is displayed on the screen.
3. View the certificate information.  
Each certificate contains fields specifying the subject, the name of the CA issuing the certificate, the algorithm used by the issuer to sign the certificate, validity period, key size and the certificate unique identifier.
4. Download the certificate.
5. Log on to the provisioning system you want to authenticate to (in this case, SAP BTP ABAP Environment) and upload the certificate to configure the certificate-based authentication there. For more information, see: [How to Create Communication Users](#)
6. Return to the Identity Provisioning UI and select the [Properties](#) tab of the provisioning system.
7. Set the [Authentication](#) property to **ClientCertificateAuthentication**.
8. Save your configuration.

## Manage Certificate Validity

To manage the certificate validity, choose your applicable approach: automatic or manual.

### Manage Validity Automatically

Enable the [Automatic Regeneration](#) option.

The expiring active certificate is set to inactive. A new certificate is automatically regenerated and set to active.

For systems validating certificates based on their subject and issuer, no further manual steps are required.

### Manage Validity Manually

1. Periodically check the validity of your active certificate.
2. When the certificate approaches its expiration date (or it has already expired), generate a new certificate, as described above. The status of this certificate will be *Inactive*.
3. Continue with downloading the inactive certificate and uploading it to the backend provisioning system.
4. Once ready, activate the second (inactive) certificate. As a result, your current active certificate will be set to inactive.
5. You can safely delete the inactive certificate.

## 1.7.10.2 Manage Certificates for Inbound Connection

Identity Provisioning handles the following tasks related to X.509 client certificates for inbound connection: importing and deleting.

Importing client certificates for inbound connection to Identity Provisioning is used in two scenarios: configuring real-time provisioning and configuring proxy systems for provisioning user data to and from a central identity management solution. Therefore, inbound client certificates can be configured for source and proxy systems only.

Certificate files with the following filename extensions are supported: `.crt`, `.pem` and `.der`.

Depending on the infrastructure/environment your Identity Provisioning tenant is running, you manage the inbound certificates in the administration console of Identity Provisioning or SAP Cloud Identity Services.

## SAP Cloud Identity Infrastructure

Bundle or standalone tenants running on SAP Cloud Identity infrastructure manage certificates for inbound connections in the SAP Cloud Identity Services admin console, where the certificate must be uploaded for the technical user of type *System*. For more information, see [Add System as Administrator](#).

If this technical user will be used to connect to Identity Provisioning proxy system, enable the [Access Proxy System API](#) permission.

## SAP BTP, Neo Environment

Bundle or standalone tenants running on SAP BTP, Neo environment manage certificates for inbound connections in the Identity Provisioning admin console.

### Prerequisites

For standalone tenants, the following requirements must have been fulfilled in SAP BTP cockpit in the consumer subaccount:

1. Create an OAuth client for Platform API and choose the [Keystore](#) and [Authorization Management](#) API options. Save the generated client credentials. For more information, see [Using Platform APIs](#)
2. Create a destination with the following properties:
  - **Name:** `IPS_MANAGE_AUTHORIZATIONS`
  - **URL:** `https://oauthservices.<neo-region-host>/oauth2/apitoken?grant_type=client_credentials`
  - **ProxyType:** `Internet`
  - **Type:** `HTTP`
  - **Authentication:** `BasicAuthentication`
  - **User:** `<client-id-platform-api-client>`
  - **Password:** `<client-secret-platform-api-client>`

### Procedure

You can import client certificates from various systems to establish a trusted inbound connection to a given source or proxy system. You are allowed to import and manage as many certificates as you need for your scenarios.

To import a certificate, proceed as follows:

1. In the Identity Provisioning admin console, select your source or proxy system.
2. Select the [Inbound Certificates](#) tab and choose [Import](#).  
The name of the imported certificate is generated following the pattern: `cert_client_<fingerprint>`.
3. View the details of the certificate and periodically check its validity.  
Each certificate contains fields specifying the subject name, the issuer, the algorithm used by the issuer to sign the certificate, validity period, key size and the fingerprint, which is the certificate unique identifier.

#### **i** Note

Delete a certificate when it is expired.

## 1.7.11 Connecting to On-Premise Systems

Set up the connection to on-premise systems, such as SAP AS ABAP, LDAP Server, Microsoft Active Directory, SAP S/4HANA On-Premise, when your Identity Provisioning bundle or standalone tenant is running on the SAP Cloud Identity Services infrastructure or SAP BTP, Neo environment.

### Related Information

[Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#)

[Connecting to On-Premise Systems in Neo Environment \[page 1516\]](#)

### 1.7.11.1 Connect to On-Premise Systems in SAP Cloud Identity Infrastructure

Set up the connection to on-premise systems when your Identity Provisioning bundle or standalone tenant is running on the infrastructure of SAP Cloud Identity Services.

### Prerequisites

- Ensure your tenant is running on SAP Cloud Identity Services infrastructure. For more information, see [Tenant Model \[page 8\]](#).
- You have installed the Cloud Connector (for SAP BTP, Cloud Foundry environment), and have done the initial configuration. For more information, see [Cloud Connector](#).

### Context

If your provisioning scenarios involve on-premise systems, this requires a separate configuration in three places:

- SAP BTP cockpit, where you subscribe to the Cloud Identity Services connectivity plan in your multi-environment subaccount.
- SAP Cloud Connector, where the connection to your multi-environment subaccount is established, and the backend (on-premise) system is defined.
- The Identity Provisioning section of SAP Cloud Identity Services administration console, where you configure the on-premise provisioning systems.



## Procedure

1. Before you start actual configuration, access the [SAP Cloud Identity Services - Tenants](#) application at the following URL: <https://iamtenants.accounts.cloud.sap/> to view the region and the type of Identity Authentication and Identity Provisioning tenants assigned to your customer ID. You will need this information later when creating multi-environment subaccount.

For more information, see [Viewing Assigned Tenants and Administrators](#).

The screenshot below illustrates a customer landscape with Identity Authentication and Identity Provisioning tenants running on their common SAP Cloud Identity Services infrastructure in one region US/Canada (in the red frame). The first pair is used for testing purposes and the second one for productive purposes.

Host	Service	Type	Default	Additional	Created	Region	Customer	Details
<b>Tenant - [redacted]</b>								
https://[redacted].accounts.ondemand.com	IAS	Test	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Aug 18, 2017	Europe (Rot)	[redacted]	...
https://ips-[redacted].dispatcher.hana.ondemand.com	IPS	Test	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Dec 11, 2019	Europe (Rot)	[redacted]	...
<b>Tenant - [redacted]</b>								
https://[redacted].accounts.ondemand.com	IAS	Test	<input type="checkbox"/>	<input checked="" type="checkbox"/>	May 27, 2021	US/Canada	[redacted]	...
https://[redacted].accounts.ondemand.com/ips	IPS	Test	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Nov 30, 2022	US/Canada	[redacted]	...
<b>Tenant - [redacted]</b>								
https://[redacted].accounts.ondemand.com	IAS	Productive	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Jul 14, 2021	US/Canada	[redacted]	...
https://[redacted].accounts.ondemand.com/ips	IPS	Productive	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Dec 11, 2022	US/Canada	[redacted]	...

2. Log on to SAP BTP cockpit and choose your global account.

If you have only one global account, you are automatically taken there. If you have multiple ones, select the global account you want to set up connection to on-premise system.

For more information on how to navigate in the SAP BTP cockpit, refer to [Navigate in the Cockpit](#).

3. Create a multi-environment subaccount in the Cloud Foundry region that maps the region of the Identity Authentication tenant (where the Identity Provisioning is also running). See the mapping table below.

Identity Authentication Region	Cloud Foundry Region
Rot (Germany) / Amsterdam (Netherlands)	Europe (Frankfurt) AWS
Germany (Frankfurt)	Europe (Frankfurt) AWS
UAE (Dubai)	Europe (Frankfurt) AWS
Saudi Arabia (Riyadh)	Europe (Frankfurt) AWS
Australia (Sydney) / Japan (Tokyo)	Australia (Sydney) AWS
China (Shanghai)	Singapore AWS

Identity Authentication Region	Cloud Foundry Region
Singapore	Singapore AWS
South Korea (Seoul)	South Korea (Seoul) AWS
Japan (Tokyo) / Japan (Osaka)	Japan (Tokyo) AWS
Brazil (São Paulo)	Brazil (São Paulo) AWS
United States (Sterling) / Canada (Toronto)	US East (VA) AWS
Canada (Toronto)	Canada (Montreal) AWS
USA West (Quincy)	US West (WA) Azure
Switzerland (Zürich)	Switzerland Azure
India (Mumbai)	India AWS

For example, customers with Identity Authentication and Identity Provisioning tenants in US/Canada region must create a Cloud Foundry subaccount in US East (VA) region.

Once you create the subaccount, you must enable Cloud Foundry for this subaccount.

#### **i Note**

If you already have a Cloud Foundry subaccount, you can use it.

- Open your subaccount, and from the left-side panel choose ► [Service Marketplace](#) ► [Integration Suite](#) ► [Cloud Identity Services](#) ►.
- Choose [Create](#), select the [connectivity](#) plan and then [Next](#).

#### **i Note**

If the connectivity plan is not present, navigate to ► [Entitlements](#) ► [Configure Entitlements](#) ► [Add Service Plans](#) ►. Search for ► [Cloud Identity Services](#) ► [connectivity plan](#) ►, add it as service plan and save your changes.

#### **i Note**

You can create only one subscription plan per subaccount. This means, you cannot have connectivity along with additional-tenant plan in the same subaccount.

- From the [Cloud Service Type](#) dropdown, choose for what type of tenant (test or productive) you want to use the subaccount.
  - [Test](#)
  - [Productive](#) - default value
- Choose [Next](#) and then [Create](#).

This creates a subscription to Cloud Identity Services connectivity plan and creates a binding to your Identity Authentication tenant and your Identity Provisioning service.

## i Note

You can have two separate Cloud Foundry subaccounts per region: one of them configured for [Productive](#) connections and the other for [Test](#) connections. While it is allowed to create multiple Cloud Foundry subaccounts per one region and per one type (test or productive), be aware that in this case only the first one where the connectivity plan was enabled will be used by Identity Provisioning.

- Now, you need to add and connect your subaccounts to the Cloud Connector. Log on to the Cloud Connector administration UI and choose [Add Subaccount](#).

For more information, see [Managing Subaccounts](#).

- Verify that the Cloud Connector is connected to your Cloud Foundry subaccount. In SAP BTP cockpit, from the left-side panel, choose [Connectivity](#) [Cloud Connectors](#) to see your up and running cloud connector tunnels.
- Return to the Cloud Connector and define the backend (on-premise) system.
- Return to your subaccount in SAP BTP cockpit and navigate to [Connectivity](#) [Destinations](#) to configure the destination for your on-premise system.

## i Note

This step is mandatory only for SAP Application Server ABAP.

For more information, see: [Create RFC Destinations](#) and [SAP Application Server ABAP → step 2](#).

- Sign in to SAP Cloud Identity Services administration console.

The URL follows the pattern: `https://<tenant_id>.accounts.ondemand.com/admin`

- In the Identity Provisioning section, add an on-premise system (source, target or proxy).

For more information, see [Supported Systems \[page 452\]](#).

- If you want to create a connection to SAP AS ABAP, from the [Destination Name](#) combo box, select the destination you have created in the cockpit and save your configurations.

If you want to create a connection to other on-premise systems, configure the connection details on the [Properties](#) tab of the given system.

- (Optional) If your Cloud Connector is configured with [Location ID](#), this location identifier must also be set for the respective on-premise system. You have the following options:

- Connectivity destination - Create it in your subaccount in the SAP BTP cockpit and provide the Location ID there.

## i Note

Using connectivity destination is mandatory only for SAP Application Server ABAP.

- `CloudConnectorLocationId` property - Configure the property in the Identity Provisioning section of SAP Cloud Identity Services administration console for all HTTP and LDAP-based systems, SSH Server (Beta) and SAP HANA Database (Beta) with `ProxyType` set to `OnPremise`.  
Alternatively, for HTTP-based systems only, you can add the Identity Provisioning `ips.http.header.<header_name>` property, where the header name is: `SAP-Connectivity-SCC-Location_ID` and the value is the Location ID. For example: `ips.http.header.SAP-Connectivity-SCC-Location_ID=<LocationID>`

- Add another provisioning system, connect it to your on-premise one, and run a provisioning job.

## Related Information

[Start and Stop Provisioning Jobs \[page 1524\]](#)

### 1.7.11.2 Connecting to On-Premise Systems in Neo Environment

Set up the connection to on-premise systems when your Identity Provisioning bundle or standalone tenant is running on SAP BTP, Neo environment.

#### Prerequisites

- Ensure your tenant is running on SAP BTP, Neo environment. For more information, see [Tenant Model \[page 8\]](#)
- You have installed the SAP Cloud Connector in your corporate environment and have completed the initial configuration. For more information, see [Cloud Connector \(Neo Environment\)](#)
- You have checked the availability of the on-premise systems in bundle tenants, For more information, see [Provisioning Systems for Bundle Tenants \[page 416\]](#)

#### Context

If your provisioning scenarios involve on-premise systems, this requires a separate configuration in three places:

- Identity Provisioning admin console, where you configure the provisioning systems, and (only relevant for bundle tenants) where necessary authorizations are granted to the admin user.
- SAP Cloud Connector, where the connection to the Identity Provisioning subaccount in SAP BTP cockpit is established and the backend (on-premise) system is defined.
- Identity Provisioning subaccount in SAP BTP cockpit, where the connection to the SAP Cloud Connector is verified and the destination for your on-premise system is defined too.

#### Procedure

##### 1. Note

Step 1 and 2 are relevant for bundle tenants. If you are using standalone tenants, start from step 3.

Log on to the Identity Provisioning admin console of your bundle tenant at: `https://ips-  
<consumer_account>.dispatcher.<region_host>/webapp/index.html`

2. Select **Security > Authorizations** and assign the *Manage On-Premise Connections* and the *Manage Destinations* roles to the admin user. For more information, see [Manage Authorizations in Neo Environment](#) → [Bundle Tenants](#) [page 1490]

Alternatively, you can add new admin users and assign those authorizations to them.

3. Connect the Cloud Connector to your Identity Provisioning subaccount. For this, you need to log on to the Cloud Connector administration UI and choose [Add Subaccount](#).
4. Provide the required information and [Save](#) your configuration.

Property	Value
<a href="#">Region</a>	Provide the region of your Identity Provisioning subaccount. You can select it from the list.
<a href="#">Subaccount</a>	<p>Provide the Identity Provisioning tenant ID.</p> <p>You can get it from the URL you use to access your tenant. For example: <code>https://ips-a12345sdf678.dispatcher.hana.ondemand.com/webapp/index.html</code>, where <b>a12345sdf678</b> is the bundle tenant ID, also known as consumer account.</p> <div data-bbox="842 987 1391 1464" data-label="Complex-Block"> <p><b>i Note</b></p> <p>It's incorrect to provide your Identity Provisioning subaccount ID here.</p> <p>Your subaccount ID is the last part of the URL you use to access your subaccount. For example: <code>https://account.hana.ondemand.com/neo/#/globalaccount/IdentityProvisioningBundle&lt;Bundle_Option&gt;/neosubaccount/aaaaa11a-b123-4567-a1b2-1234567890123bb</code>, where <b>aaaaa11a-b123-4567-a1b2-1234567890123bb</b> is your subaccount ID.</p> </div>
<a href="#">Display Name</a>	(Optional) Add a name for the subaccount.
<a href="#">Subaccount User</a>	Add your subaccount (S-User) username. This is the admin user with the assigned authorizations in step 2.
<a href="#">Password</a>	Add your subaccount (S-User) password.
<a href="#">Location ID</a>	<p>(Optional) Provide a Location ID that identifies the location of this Cloud Connector for the subaccount.</p> <p>The Location ID is used to distinguish between multiple Cloud Connectors connected to one subaccount and serves as a "router" for the respective destinations created for your on-premise systems.</p>

Property	Value
	<p>If Location ID is configured for a Cloud Connector, it needs to be configured in the destination of the respective on-premise system too. Thus, the Location ID in the destination will identify the relevant Cloud Connector over which the connection will be opened.</p> <p>The Location ID must be unique per subaccount. For example: <b>Sydney</b></p>
<i>Description</i>	(Optional) Provide a description for the subaccount.

- Verify that the Cloud Connector is connected to your Identity Provisioning subaccount. For this, you need to log on to your Identity Provisioning subaccount in SAP BTP cockpit and navigate to ► [Connectivity](#) ► [Cloud Connectors](#) ►.
- Your subaccount URL follows the pattern: `https://account.<region-host>/cockpit#/acc/<ips-tenant-ID>`
- For example: `https://account.eu2.hana.ondemand.com/cockpit#/acc/abc123456`
- Return to the Cloud Connector and define the backend (on-premise) system. For this, you need to choose [Cloud To On-Premise](#) option and provide all the required information in order to map the virtual system to the internal system.
  - Return to your Identity Provisioning subaccount and configure the destination for your on-premise system. For this, you need to navigate to ► [Destinations](#) ► [New Destination](#) ►.
  - Save your configuration.

## Next Steps

Now that you've configured the connection to the on-premise system using the Cloud Connector, you can proceed with setting up your provisioning scenario in the Identity Provisioning UI.

The following is an example scenario which is applicable for all bundle options:


- Configure an on-premise system as a source system. For example: SAP S/4HANA On-Premise  
Note that when you select a destination (defined in the subaccount), it will take up to 5 minutes for it to appear in the Identity Provisioning UI.
- Configure an SAP cloud service as a target system. For example: Identity Authentication
- Run a provisioning job.

## 1.7.12 Manage Full and Delta Read

When you set up your systems and start a scheduled provisioning task, the standard behavior of the process reads all the entities from the source system. This mode prevents data loss and always keeps your target system synchronized with the source. However, it may take a long time for every job to be executed.

### Context

Delta read is a concept for optimizing the amount of data retrieved from the source system. Delta read is much faster, but sometimes might have limitations. In order for a source system to support delta read mode, its API should allow the implementation of this feature.

For example, the [Microsoft Active Directory](#) source system uses the **uSNChanged** attribute. For more information, see [Microsoft: Polling for Changes Using USNChanged](#) .

The main difference between delta and full read is:

- **Delta read** – only modified data is read from the source system and triggered to the target one. Modified data means: new entities and updates on existing entities. Entities deleted from the source system will not be deleted from the target. They can be deleted only during a **full read** job.
- **Full read** – all entities (new, updated, deleted, and existing unchanged ones) are read and checked every time a provisioning job is triggered to the target system.

To keep source and target systems completely synchronized, you can use the **Resync** type of provisioning job.

#### → Tip

We recommend that you enforce full reads from time to time if the connector is in delta read mode. To achieve this, you need to set up the following source system property: `ips.full.read.force.count`. For example, `ips.full.read.force.count = 10` will result in alternating full reads after every 10 delta reads are performed.

This property only impacts scheduled runs; manually triggered runs are ignored. In case it is not set, only delta read jobs will be executed.

#### → Remember

When the Identity Provisioning reads entities from a source system for the first time, it always triggers a **full read** job. If the job is successful, the service can then continue with delta read jobs (if such are activated). During a delta read job, the service reads only the entities that are new or have been modified after the last successful job.

Below are listed all source systems that currently support **delta read** mode.

## Supported Systems

System Type	Details
<i>SAP SuccessFactors</i>	<p>Default mode: <i>Delta read</i></p> <p>You can switch to full read, if you set up the relevant property:  <code>ips.delta.read = disabled</code></p>
<i>SAP SuccessFactors Learning</i>	<p>Default mode: <i>Delta read</i></p> <p>You can switch to full read, if you set up the relevant property:  <code>ips.delta.read = disabled</code></p>
LDAP-based systems	
<i>Microsoft Active Directory</i>	<p>Default mode: <i>Full read</i></p> <p>You can switch to delta read, if you set up the relevant property:  <code>ips.delta.read = enabled</code></p> <p>Bear in mind the following specifics and limitations:</p> <ul style="list-style-type: none"> <li>• Make sure that the service user, which is used in the AD destination, has a <b>Domain Admin</b> role, otherwise the connector won't be able to extract any data from the recycle bin.</li> <li>• Due to the <i>linked attributes</i> concept of AD, there is a limitation in the Microsoft Active Directory read connector, when performing in delta read mode. We recommend that you enforce full reads periodically in order to avoid data loss. See: <a href="#">Microsoft: Linked Attributes</a> ➡</li> <li>• You need to set limitations about which particular attributes to be read. For this purpose, set the properties <code>ldap.user.attributes</code> and <code>ldap.group.attributes</code> and add <b>uSNChanged</b> to the attributes list. Otherwise, the provisioning job will run in <i>full read</i> mode.</li> <li>• If an entity is moved outside the base path (another directory context), the connector won't recognize this change during delta read.</li> </ul>
SCIM-based systems	



System Type	Details
<a href="#">Identity Authentication</a>	Default mode: <a href="#">Full read</a>
<a href="#">Local Identity Directory</a>	You can switch to delta read, if you set up the relevant property:
<a href="#">SAP Central Business Configuration</a>	<code>ips.delta.read</code> = <b>enabled</b>
<a href="#">SAP Data Custodian</a>	<div><b>i Note</b></div> <p>When using SAP Central Business Configuration and Identity Directory SCIM API (in short, SCIM API version 2), delta read mode is only supported for user resources.</p>
<a href="#">SAP CPQ</a>	
<a href="#">SAP Advanced Financial Closing</a>	
<a href="#">SCIM System</a>	
(General SCIM system, if fulfills the API requirements)	<p>For delta read of resources (users and groups), bear in mind the following API requirements:</p> <ul style="list-style-type: none"> <li>The system API should return <b>lastModified</b>, which is a subattribute of the <b>meta</b> attribute. The <b>lastModified</b> subattribute denotes the most recent date and time when the resource details were updated at the service provider. See: <a href="#">SCIM: Common Attributes</a> ➡</li> <li>The system API has to also support filtering by the <b>lastModified</b> attribute, and the system should support the <b>gt</b> operator in filter expressions. See: <a href="#">SCIM: Filtering</a> ➡</li> </ul>

## Related Information

[SAP SuccessFactors \[page 635\]](#)

[SAP SuccessFactors Learning \[page 649\]](#)

[Microsoft Active Directory \[page 676\]](#)

[Identity Authentication \[page 453\]](#)

[Identity Directory \[page 1567\]](#)

[SCIM System \[page 695\]](#)

## 1.7.13 Manage Deleted Entities

Manage deletion of entities (users or groups) in the target system after they have been deleted from the source system.

Scenario	Solution
<b>Scenario 1</b>  An entity exists both in the source and the target system. <ol style="list-style-type: none"><li>1. You run a provisioning job for the first time. As a result, Identity Provisioning reads this entity from the source and updates it on the target system.</li><li>2. You delete the relevant entity from the source system.</li><li>3. You run another provisioning job, which finishes successfully. However, the service recognizes the relevant entity as a "previously existed one" and <b>does not delete</b> it from the target.</li></ol>	<p>To delete entities from a target system after they have been deleted from the source system, you need to set the following property:</p> <p><b><code>ips.delete.existedbefore.entities</code></b> = <b><code>true</code></b> in the target system. This must be done <b>before</b> the job to delete those entities from the target system is executed.</p> <div><p>→ Recommendation</p><p>The following sequence of steps is recommended for synchronizing deletion of entities between source and target systems, as in <b>Scenarios 1, 2 and 3</b>:</p><p>You have run successful provisioning jobs (<a href="#">Read</a> or <a href="#">Resync</a>) between the systems.</p><ol style="list-style-type: none"><li>1. Delete an entity from the source system.</li><li>2. On the <a href="#">Properties</a> tab of the target system, add the <b><code>ips.delete.existedbefore.entities</code></b> property and set its value to <b><code>true</code></b>.</li><li>3. Run a provisioning job.</li><li>4. Verify that the relevant entity has been deleted from the target system.</li></ol></div>
<b>Scenario 2</b>  An entity does not exist in either system (neither source, nor target). <ol style="list-style-type: none"><li>1. You run provisioning jobs (<a href="#">Read</a> or <a href="#">Resync</a>) between the systems.</li><li>2. You add this entity to the source system.</li><li>3. The same entity is added (manually or via script) to the target system.</li><li>4. You run a new provisioning job. As a result, Identity Provisioning reads this entity from the source and updates it in the target system.</li><li>5. You delete the relevant entity from the source system.</li><li>6. You run another provisioning job, which finishes successfully. However, the service recognizes the relevant entity as a "previously existed one" and <b>does not delete</b> it from the target.</li></ol>	<p>If the property is set <b>afterwards</b>, entities recognized as "previously existed ones" cannot be deleted from the target system anymore. In this case, you need to delete them from the target system (for example, manually or via script).</p> <p>The <b><code>ips.delete.existedbefore.entities</code></b> is an optional property which can be set on every target system. You can use it to control whether recognized entities as "previously existed ones" should be deleted from the target system.</p> <p>This is important for security and legal reasons in cases when users (for example, employees) are no longer active in the source system, and their availability and permissions must be removed from the relevant target system(s).</p> <p>For more information about this property, see: <a href="#">List of Properties [page 94]</a>, where you can search it by <a href="#">Name</a> or use the general table search.</p>

**Scenario 3**

An entity exists in the source system only.

1. You run at least one provisioning job.  
As a result, Identity Provisioning reads this entity and creates it in the target system.
2. You reset one of these systems. See: [Reset Identity Provisioning System \[page 1542\]](#)
3. You run a new provisioning job.  
As a result, Identity Provisioning reads this entity from the source system (but is not "aware" of it, that is, it behaves like reading it for the first time) and makes a full update of it in the target system.
4. You delete the relevant entity from the source system.
5. You run another provisioning job, which finishes successfully.  
However, the service recognizes the relevant entity as a "previously existed one" and **does not delete** it from the target.

**Scenario 4**

An entity exists both in the source and the target system. (It has **not** been created on the target by the Identity Provisioning service.)

Conditions or expressions, such as (*ignore* or *skipOperations*), are not set in the target transformation.

1. You run a successful *Read* job. As a result, Identity Provisioning updates the existing entity on the target system.
2. You delete this entity from the source system.
3. You run a provisioning job, which finishes with error.  
As a result, the relevant entity has not been deleted from the target system.
4. In the job log, you see that there are failed entities (users or groups) on the source system. That means, the job has failed trying to read them from the source.

1. Resolve the failed entities in the source system.
2. On the *Properties* tab of the target system, add the **ips.delete.existedbefore.entities** property and set its value to *true*.
3. Run a successful *Read* job between the systems.
4. Verify that the relevant entity has been deleted from the target system.

**→ Tip**

Even if the job fails due to errors on the target system, if the read from the source is successful, the service will still delete the entity from the target.

Scenario	Solution
<p><b>Scenario 5</b></p> <p>An entity exists in the source system and has been provisioned to the target by the Identity Provisioning service.</p> <p>Conditions or expressions, such as (<i>ignore</i> or <i>skipOperations</i>), are not set in the target transformation.</p> <ol style="list-style-type: none"> <li>1. You delete this entity from the source system.</li> <li>2. You run a provisioning job, which finishes with error. As a result, the relevant entity has not been deleted from the target system.</li> <li>3. In the job log, you see that there are failed entities (users or groups) on the source system. That means, the job has failed trying to read them from the source.</li> </ol>	<ol style="list-style-type: none"> <li>1. Resolve the failed entities in the source system.</li> <li>2. Run a successful <i>Read</i> job between the systems.</li> <li>3. Verify that the relevant entity has been deleted from the target system.</li> </ol> <div> <p>→ Tip</p> <p>Even if the job fails due to errors on the target system, if the read from the source is successful, the service will still delete the entity from the target.</p> </div>
<p><b>Scenario 6</b></p> <p>An entity exists in the source system and has been provisioned to the target by the Identity Provisioning service.</p> <p>Conditions or expressions, such as (<i>ignore</i> or <i>skipOperations</i>), are not set in the target transformation.</p> <ol style="list-style-type: none"> <li>1. You delete an entity from the source system.</li> <li>2. You run a <b>delta read</b> job, which finishes successfully. However, the relevant entry has not been deleted from the target system. That's because <i>delta read</i> jobs do not take deleted users into consideration. To learn more, see: <a href="#">Manage Full and Delta Read [page 1519]</a></li> </ol>	<ol style="list-style-type: none"> <li>1. On the <i>Properties</i> tab of the source system, set the <code>ips.delta.read</code> property to <i>false</i>. Alternatively, you can wait for the next scheduled <i>full job</i> to start (if it's coming soon), according to the number you have set for property <code>ips.full.read.force.count</code>.</li> <li>2. Run a new provisioning job (or wait for it to run automatically). It will be a <i>full read</i> job.</li> <li>3. Verify that the relevant entity has been deleted from the target system.</li> <li>4. (Optional) If you want to continue running <i>delta read</i> jobs, go to the <code>ips.delta.read</code> property and set it back to <i>true</i>.</li> </ol>

## 1.7.14 Start and Stop Provisioning Jobs

You can start and stop a provisioning job from the Identity Provisioning user interface (UI) or from an API client by using the Identity Provisioning tenant admin API.

### Prerequisites

- Your source and target systems are configured and enabled.
- (Optional) You have run a *Simulate* and/or a *Validate* job before you run the actual provisioning job to verify that Identity Provisioning configurations produce the desired result in the target systems.

## Job Types

The Identity Provisioning service provides the following types of provisioning jobs:

Run From	Job Type	Real Provisioning
Admin console	<a href="#">Read Job</a> - Reads all entities from the source system and provisions only new or updated entities to the target system. If the job is run in <b>delta read</b> mode, it reads and provisions only new or updated entities in the source system.  See <a href="#">Read Provisioning Job [page 1526]</a>	Yes
	<a href="#">Resync Job</a> - Reads all entities from the source system and provisions all entities to the target system.  See <a href="#">Resync Provisioning Job [page 1528]</a>	
	<a href="#">Simulate Job</a> - Estimates the number of entities that will be created, updated, deleted or skipped in the target system. Provides the expected results of a resync job without modifying the target system.  See <a href="#">Simulate Provisioning Jobs [page 1528]</a>	No
	<a href="#">Validate Job</a> - Verifies how entities (users and groups) would be mapped from source to target systems without modifying them.  See <a href="#">Validate Provisioning Jobs [page 1529]</a>	
API client	Use the Identity Provisioning tenant admin API to run a provisioning job from an API client. The API is available on the SAP Business Accelerator Hub.  See <a href="#">Run Provisioning Jobs via API [page 1532]</a>	Yes


## Start a Job

To run a job, select a source system and choose [Jobs](#) > [<Job\\_Type>](#) > [Run Now](#).

## Schedule a Job

To schedule a job run, select a source system and choose [Jobs](#) > [Read Job](#) > [Schedule](#).

## Stop a Job

To stop a running job, select a source system and choose the  [Stop Job](#) button in the [Action](#) column.

## Related Information

[Monitor Provisioning Job Logs \[page 1594\]](#)

[Manage Provisioning Job Logs \[page 1600\]](#)

## 1.7.14.1 Read Provisioning Job

The [Read](#) job reads all entities from the source system and provisions only new or updated entities to the target one. If the job is run in **delta read** mode, it reads and provisions only new or updated entities in the source system.

### i Note

A Read job checks only for changes in the source system. If there have been changes in the target system, they are not affected by the job.

Although the name of the job implies reading users, a Read job could also lead to deleting users from the target system. This could happen if any of the following changes occur after you have run a provisioning job (Read or Resync):

- A user has been deleted in the source system. As a result, on the next run of the Read job, the deleted user won't be read from the source system and will be deleted in the target system.
- A user filter is applied in the source system. As a result, on the next run of the Read job, the users not matching the filtering criteria will be deleted in the target system.
- A condition is applied either in the source or in the target system. As a result, on the next run of the Read job, the users not matching the condition criteria will be skipped or deleted in the target system.
- The mapping of the targetVariable `entityIdSourceSystem` or `entityIdTargetSystem` has been changed.

### ❖ Example

You have a source system with users A, B, C and D.

1. Run a Read Job.  
Users A, B, C and D are created in the target system.
2. In the source system, update user C, delete user D and add a new user E.
3. Run a second job.
  - If the second job is a Read Job: It reads users A, B, C and E from the source system and then creates user E, updates user C, and deletes user D in the target system.
  - If the second job is a Delta Read job: It reads users C and E from the source system and then creates user E and updates user C.
  - If the second job is a Resync job: It reads users A, B, C and E from the source system and then creates user E, updates users A, B and C, and deletes user D.

## Run Now

To run a read job, select a source system and choose ► [Jobs](#) ► [Read Job](#) ► [Run Now](#) ►. This starts a read job immediately.

## Schedule

To schedule a read job, select a source system and choose ► [Jobs](#) ► [Read Job](#) ► [Schedule](#) ►. This runs the job at the scheduled time period.

Option	Description
<a href="#">ON/OFF</a>	<p>Turn the job scheduler on or off:</p> <ul style="list-style-type: none"><li>• <a href="#">ON</a> - enables the job scheduler. After that, jobs run at the scheduled time period. This option does not run the job immediately.</li><li>• <a href="#">OFF</a> - stops the job scheduler. After that, jobs are stopped from running at the scheduled time. This option does not stop/pause a running job.</li></ul>
<a href="#">Run job every (minutes)</a>	<p>Schedule how often a read job to be run. The number must be larger than <b>30</b> (minutes). This option sets the time period but does not run the job immediately. Thus, after you set a scheduled period, turn <a href="#">ON</a> the job scheduler.</p> <p>Use cases:</p> <ul style="list-style-type: none"><li>• If the scheduled job is finished <b>before</b> the next time point, it will start again on the upcoming time point as scheduled, at the precise second.</li><li>• If the scheduled job is finished <b>after</b> the next time point, it will skip the relevant upcoming time point(s) and will start again on the next available one, at the precise second.</li><li>• If <b>during</b> a running scheduled job, you click <a href="#">Run Now</a>, this will trigger a manual job, which will start immediately after the current scheduled job has finished. The next scheduled job will start at the upcoming regular time point, at the precise second.</li></ul> <div><p>→ Remember</p><p>Before you schedule a job, make sure it's not been paused (that is, the job scheduler has not been turned off). Otherwise, the job will not be executed.</p></div>
<a href="#">Run job on a specific day of the week and time:</a>	<p>Specify the day and time to run the job. You must select at least one day of the week and the exact time. Day and time fields cannot be empty.</p> <p>The job is run in customer's time zone.</p> <div><p>i Note</p><p>This functionality is available only for Identity Provisioning bundle and standalone tenants running on SAP Cloud Identity infrastructure.</p></div>

## 1.7.14.2 Resync Provisioning Job

The [Resync](#) job reads all entities from the source system and provisions all entities to the target system.

This job performs a full replace of entities in the target system with entities from the source system. You can use it to fix inconsistent data between both systems. For example, when an entity has been changed or deleted in the target system only.

A Resync job overwrites changes in the target system. As a result, the data between the source and the target system becomes consistent again.

To run a resync job, select a source system and choose [Jobs](#) > [Resync Job](#) > [Run Now](#) . This starts a resync job immediately.

### Note

Normally, you don't schedule a [Resync Job](#) as it is expected to be run on demand.

## 1.7.14.3 Simulate Provisioning Jobs

You can simulate a provisioning job before you actually run it.

### Prerequisites

### Note

This functionality is available only for Identity Provisioning tenants running on SAP Cloud Identity infrastructure.

The source system where you run the job must be enabled.

### Context

Simulating a provisioning job allows you to test your Identity Provisioning configurations and see whether they produce the desired result in the target system. In case the result is not what you have expected, you can identify the wrong configurations and correct them before you run the actual provisioning job.

The simulate job reads the data from the source system, applies the source and target system transformation and provides the expected results of a resync job without actually modifying the target system. The simulation is based on the Identity Provisioning operational data from previously provisioned entities using the same source and target system configurations.

Run this job whenever you change filtering properties, apply conditions or modify attribute mappings in the transformation code of a source and/or target system because these changes affect the data you want to



provision. As a result, in the [Job Execution Logs](#) screen, you will see an estimation of the numbers of users and groups that will be created, updated, deleted or skipped in the target system.

Note that, if some of the entities already exists on the target system, the simulation won't detect them. In such cases during the real provisioning Identity Provisioning will detect them and try to update them, so they may appear as updated entities. The simulation can detect some wrong attribute mappings as missing attributes that are defined as required in the transformation. However, it cannot predict errors that may occur on the target system as duplicate identifiers or not supported values.

Proceed as follows:

## Procedure

1. Open the source system and choose the [Jobs](#) tab.
2. Choose [Run Now](#) for the [Simulate Job](#).
3. Open the [Job Execution Logs](#) and verify the provisioned entities.

## Results

If the number of created, updated, deleted or skipped entities is what you have expected, run the read or resync job.

If the number of created, updated, deleted or skipped entities is not what you have expected, correct your configurations and run the simulate job again.

## Related Information

**Blog Post:** [Simulate it until you make it! Try out the Identity Provisioning job that tests your configuration.](#) 

## 1.7.14.4 Validate Provisioning Jobs

You can validate a provisioning job before you actually run it.

## Prerequisites

- You have enabled the source system where you want to run the validate job from.
- You have created separate CSV input files for validating users and groups.

## i Note

This functionality is available only for Identity Provisioning tenants running on SAP Cloud Identity infrastructure.

## Context

The validate job allows you to test how entities (users and groups) would be mapped from source to target systems before you run the actual provisioning job. You can use it to validate both - default and modified transformations, and see what would be the expected result in the target system.

Like the simulate job, the validate job predicts results in the target system without modifying it. However, unlike the simulate job (which estimates the number of entities that will be created, updated, deleted or skipped), the validate job verifies the content of the entities. For example, you can use it in the following cases:

- You want to run your first provisioning job from system A to system B and see what will be the result when the default transformations are applied.
- You have modified the **sourcePath** or the **targetPath** attributes in the transformations and you want to see how they will be mapped in the target system. For example, instead of populating the username in SAP Analytics Cloud target system with the email, you want to provision the username itself:

### Default Mapping

#### ≡ Code Syntax

```
{
  "sourcePath": "$.emails[0].value",
  "targetPath": "$.userName"
},
```

### Changed Mapping

#### ≡ Code Syntax

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName"
},
```

- You want to test if conditions are met or if all required user and group attributes are available

To run a validate job, you need to import one or two CSV files - one for users and/or one for groups. Each file must contain a maximum of 10 users or groups with attributes as defined in the source system. The attribute names must use the JSONPath dot-notation. For example: **emails[0].value** attribute indicating the value of the first email within an email array, or **name.familyName** - a complex **name** attribute containing a **familyName** sub-attribute. If the file has more than 10 users or groups, the job will validate the first 10 and will ignore the rest.

You can create a CSV file using a text editor, for example Notepad. Make sure the number of fields match the number of headers in the file. Otherwise, you will get an error when importing the file and won't be able to proceed until you correct it.

## ❖ Example

The following example illustrates that the number of fields does not match the number of headers (columns 1 to 5) at row 3.

### Code Syntax

```
col1,col2,col3,col4,col5  
A,B,,D,E  
L,M,N,,  
P,Q,
```

For more information on how to create the CSV files and end-to-end examples, see the blog post [After simulation, try out validation. Identity Provisioning closes the loop with a fresh new test job.](#)

The validate job is triggered manually. There is no option to schedule it. Proceed as follows:

## Procedure

1. Open the source system and choose the *Jobs* tab.
2. For *Validate Job*, choose *Run Now*.
3. In the *Import Entities* dialog box, browse for and select the CSV files for testing users and groups.  
You can import and validate one of the files or both.
4. Choose *Validate*.

## Results

The result of a validate job is not displayed in the Job Logs, as it is with the read, resync and simulate jobs. It is provided in a downloadable ZIP file. Each administrator who runs the validate job, can see only his or her test results.

The ZIP file contains one or more CSV files, and a trace.log file in case the validation job identified failed or skipped entities. You can expect the following zipped files: one for each of the entities (users and groups) you have tested, and one for each of the target systems connected to the source system where you run the job from.

For example, if you have tested users and groups from source system A, which is connected to three target systems B, C, and D, expect a ZIP file with six CSV files.

If the result of a validate job is not what you have expected or finished with errors and skipped entities, you can identify where the wrong configurations or missing values come from and correct them.

## 1.7.14.5 Run Provisioning Jobs via API

You can run a provisioning job via API requests.

### Prerequisites

You need a technical user with *Access Identity Provisioning Tenant Admin API* permission assigned. For more information, see [Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#).

### Context

Use the Identity Provisioning tenant admin API to run a provisioning job from an API client. The API is available on the SAP Business Accelerator Hub: [SAP Cloud Identity Services](#) ► [Identity Provisioning Service](#) ► [API Reference](#) ► [Jobs](#) . The URL for accessing the Tenant Admin API follows the pattern: `https://<ias-tenant-host>/ips/service/publicapi/v1/startJob/{SourceSystemId}/jobs/{JobType}`, where:

- `<SourceSystemId>` is the ID of the source system, displayed at the end of the system URL in the SAP Cloud Identity Services administration console.
- `<JobType>` is **READ** or **RESYNC**.

#### i Note

This functionality is available only for Identity Provisioning tenants running on SAP Cloud Identity infrastructure.

## 1.7.15 Handle Rate Limits

Identity Provisioning APIs implement rate limits to control the number of incoming requests for a given time.

If an external system sends requests to Identity Provisioning proxy or real-time APIs and gets the 429 status code *Too Many Requests*, this means that the specified rate limits of the service are exceeded. As a result, Identity Provisioning rejects the requests for a period of time. Rate limits help avoid overloading, misuse of APIs and performance issues.

Identity Provisioning Rate Limits

Value	Description
150	Allowed requests per minute. When the limit is reached, the requests are slowed down.

Value	Description
200	Maximum requests per minute. When the limit is reached, further requests are immediately rejected.

The Identity Provisioning rate limits are enforced at tenant level to the total number of incoming requests (that is, incoming calls for real-time provisioning and proxy systems).

For more information on how Identity Provisioning handles failed operations when the service sends requests to external systems, see [Handle Failed Operations \[page 1533\]](#).

## 1.7.16 Handle Failed Operations

In certain cases, you can set a retry for a failed operation due to an occurred exception.

If an entity operation ([get](#), [create](#), [update](#), or [delete](#)) fails due to an occurred exception (rate limit, bad gateway, missing authorization, or timeout), you can set it for retry by the Identity Provisioning. You can specify a number of retries by setting the property `ips.failed.request.retry.attempts`. You can also specify a time interval (in seconds) between the retries via the property `ips.failed.request.retry.attempts.interval`. For more information, see [List of Properties \[page 94\]](#).

Based on the system you use, the type of failed operation, and the occurred exception, the option to retry is possible in the following cases:

Identity Provisioning Retry Support

Exception	Write Operation via job or Proxy	Read Operation via job	Read Operation via Proxy
<i>Timeout exception</i>	All connectors	All connectors	The retry is not supported.

**i Note**

The retry is supported only for full read.

Exception	Write Operation via job or Proxy	Read Operation via job	Read Operation via Proxy
<i>Too many requests (429)</i>	<ul style="list-style-type: none"> <li>Identity Authentication</li> <li>Microsoft Azure Active Directory</li> <li>SAP Analytics Cloud</li> <li>SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>SAP Sales Cloud and SAP Service Cloud</li> <li>SAP SuccessFactors version 2</li> <li>SAP Jam Collaboration</li> <li>SCIM System</li> </ul>	<ul style="list-style-type: none"> <li>Cloud Foundry UAA server</li> <li>Identity Authentication</li> <li>Local Identity Directory</li> <li>Sales Cloud – Analytics &amp; AI</li> <li>SAP Advanced Financial Closing</li> <li>SAP Analytics Cloud</li> <li>SAP Ariba Applications</li> <li>SAP BTP Account Members (Neo)</li> <li>SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>SAP Business Network</li> <li>SAP Build Work Zone, advanced edition</li> <li>SAP Build Work Zone, standard edition</li> <li>SAP Central Business Configuration</li> <li>SAP Commerce Cloud</li> <li>SAP Commissions</li> <li>SAP Concur version 2</li> <li>SAP CPQ</li> <li>SAP Data Custodian</li> <li>SAP Enterprise Portal</li> <li>SAP Fieldglass</li> <li>SAP Field Service Management</li> <li>SAP Jam Collaboration</li> <li>SAP Sales Cloud and SAP Service Cloud</li> <li>SAP SuccessFactors Learning</li> <li>SAP SuccessFactors version 2</li> <li>SAP S/4HANA for procurement planning</li> <li>SCIM System</li> </ul>	<ul style="list-style-type: none"> <li>Cloud Foundry UAA server</li> <li>Identity Authentication</li> <li>Local Identity Directory</li> <li>Sales Cloud – Analytics &amp; AI</li> <li>SAP Advanced Financial Closing</li> <li>SAP Analytics Cloud</li> <li>SAP Ariba Applications</li> <li>SAP BTP Account Members (Neo)</li> <li>SAP BTP XS Advanced UAA (Cloud Foundry)</li> <li>SAP Business Network</li> <li>SAP Build Work Zone, advanced edition</li> <li>SAP Build Work Zone, standard edition</li> <li>SAP Central Business Configuration</li> <li>SAP Commerce Cloud</li> <li>SAP Commissions</li> <li>SAP Concur version 2</li> <li>SAP CPQ</li> <li>SAP Data Custodian</li> <li>SAP Enterprise Portal</li> <li>SAP Fieldglass</li> <li>SAP Field Service Management</li> <li>SAP Jam Collaboration</li> <li>SAP Sales Cloud and SAP Service Cloud</li> <li>SAP SuccessFactors Learning</li> <li>SAP SuccessFactors version 2</li> <li>SAP S/4HANA for procurement planning</li> <li>SCIM System</li> </ul>
		<b>i Note</b> The retry is supported only for full read.	<b>i Note</b> The retry is supported for single entity read by ID and for full read.

Exception	Write Operation via job or Proxy	Read Operation via job	Read Operation via Proxy
<i>Bad gateway (502)</i>	<ul style="list-style-type: none"> <li>SAP Build Work Zone, standard edition</li> <li>SAP BTP XS Advanced UAA (Cloud Foundry)</li> </ul> <div> <p><b>i Note</b></p> <p>For SAP Build Work Zone, standard edition and SAP BTP XS Advanced UAA (Cloud Foundry) is supported retry only for patch operation.</p> </div> <ul style="list-style-type: none"> <li>SAP Sales Cloud and SAP Service Cloud</li> </ul>	SAP Sales Cloud and SAP Service Cloud	SAP Sales Cloud and SAP Service Cloud
<i>Forbidden (403)</i>	SAP Analytics Cloud	The retry is not supported.	The retry is not supported.

## Related Information

[List of Properties \[page 94\]](#)

## 1.7.17 Configure Identity Provisioning in SAP Cloud Identity Services Administration Console

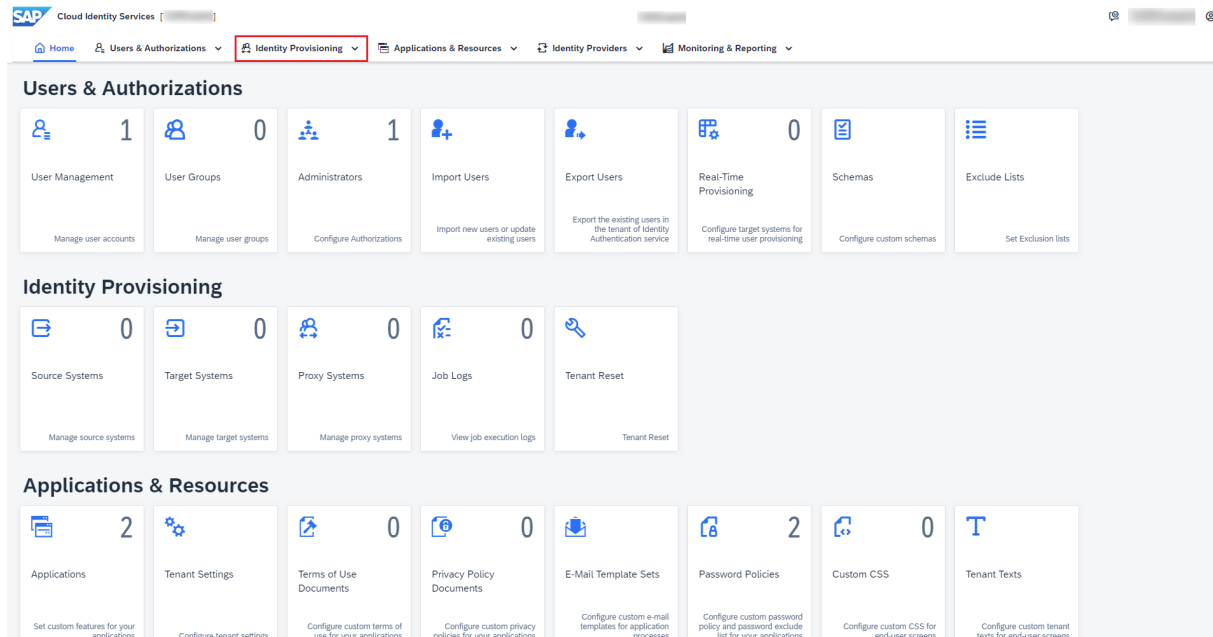
Administrators of Identity Provisioning tenants running on SAP Cloud Identity infrastructure can configure and work with the provisioning functionality in the administration console of SAP Cloud Identity Services.

## Prerequisites

- The Identity Provisioning service must be enabled for your Identity Authentication tenant.
- The [Manage Identity Provisioning](#) role must be enabled for the Identity Authentication and Identity Provisioning administrator.

## Context

Identity Provisioning is embedded in the SAP Cloud Identity Services administration console. The entire provisioning functionality can be accessed there in the navigation area under [Identity Provisioning](#). Sharing one administration console is a step further in tightening the SAP Cloud Identity Services integration.



1. Sign in to the administration console of SAP Cloud Identity Services.  
The URL follows the pattern: `https://<ias-host>/admin`.

### Note


If your tenant URL is `https://<ias-host>/ips`, you will be redirected to `https://<ias-host>/admin`.

2. In the navigation area, select [Identity Provisioning](#) and proceed with your configuration.

## 1.7.18 Migrate Identity Provisioning Bundle Tenant

Migrate your Identity Provisioning bundle tenant on SAP BTP, Neo environment to the infrastructure of SAP Cloud Identity Services.

## Prerequisites

- Ensure you have a bundle tenant running on SAP BTP, Neo environment. Standalone tenants cannot be migrated through the wizard option described below.  
For more information on how to proceed with standalone tenants on Neo environment, refer to the SAP Note [3278189](#) .



- Ensure you have an admin user on the target Identity Authentication tenant.
- Ensure that no provisioning jobs are running before you start the migration. Stop manually triggered jobs and pause the scheduled ones.
- If you use connectivity destination for any of your systems, do not modify them while the migration is running.

## Context

Any administrator with *Manage Identity Provisioning* permission can trigger the migration of Identity Provisioning bundle tenants on SAP BTP, Neo environment to the infrastructure of SAP Cloud Identity Services. Sharing the same infrastructure with Identity Authentication brings a number of benefits as described in [Tenant Infrastructure \[page 10\]](#).

As a result of a successful migration, you will have a tenant that includes Identity Authentication and Identity Provisioning services. You will use the `https://<ias-tenant-host>/admin` tenant URL to access the common administration console of SAP Cloud Identity Services.

During migration, your Identity Provisioning tenant will be disabled. Other administrators of this tenant won't be able to perform any operation or system modification until it completes.

### i Note

The migration process might take considerable time to complete depending on the amount of data you want to migrate. You will be updated regularly about its progress in the Identity Provisioning UI.

A wizard guides you through the migration process. First, you choose the target Identity Authentication tenant to which you want to migrate your Identity Provisioning tenant. Then, you select the provisioning systems - source, target and proxy, regardless of their status (enabled or disabled). The process moves the system configurations, such as properties, destinations, credentials and transformations, to the tenant in the new infrastructure. Provisioning job logs are not migrated. They are retained in your Neo tenant according to the period of time (7, 14 or 30 days) you have configured. The migration job log is retained for 30 days.

### i Note

Following a successful migration, you will have access to your Identity Provisioning tenant on SAP BTP, Neo environment for 30 days. After that time, the tenant is offboarded and cannot be restored.

## Procedure

1. Log on to your Identity Provisioning tenant and select *Tenant Migration*.
2. Choose *Migrate*.

This opens the tenant migration wizard.

On this step, you start configuring the data you want to migrate. The migration itself is triggered when you choose *Finish* on the last step of the wizard.

### **i** Note

From now on, if you choose [Cancel](#) on any of the steps below, configurations already made through the wizard won't be saved. Next time you choose [Migrate](#), you'll start over with your configurations.

3. Select the target Identity Authentication tenant to which you want to migrate your Identity Provisioning tenant.

The dropdown displays the available Identity Authentication tenants for your Identity Provisioning tenant on Neo environment. If you want to select a different tenant, open an incident to BC-IAM-IPS component and request it.

Once you make your selection, the following tenant specific information is displayed: name and ID, region and host, the date when the tenant was created and its initial administrator. Depending on the value of [Identity Provisioning already exists](#) field, you could expect the following:

- If set to [true](#), this means that the Identity Provisioning service is already enabled for the selected Identity Authentication tenant in the SAP Cloud Identity infrastructure and your data will be migrated there.
- If set to [false](#), this means that there is no Identity Provisioning service enabled for the selected Identity Authentication tenant in the SAP Cloud Identity infrastructure. The service will be enabled and your data will be migrated there.

Once you view the details, choose [Next](#).

### **i** Note

You are allowed to migrate the Identity Provisioning tenant without migrating its provisioning systems (step 4,5 and 6 below). If migration finishes with errors, you can start over. If migration is successful, you cannot trigger the process again. You can only proceed with exporting the provisioning systems from your Neo tenant and importing them into the new tenant on SAP Cloud Identity infrastructure.

4. Select the source systems you want to migrate and choose [Next](#).

Initially, up to 10 systems are displayed. Be careful when you choose the [select all](#) checkbox at this point, as it will only select the initially displayed systems. Keep expanding the list until all your source systems are displayed.

5. Select the target systems you want to migrate and choose [Next](#).

Initially, up to 10 systems are displayed. Be careful when you choose the [select all](#) checkbox at this point, as it will only select the initially displayed systems. Keep expanding the list until all your target systems are displayed.

6. Select the proxy systems you want to migrate and choose [Next](#).

Initially, up to 10 systems are displayed. Be careful when you choose the [select all](#) checkbox at this point, as it will only select the initially displayed systems. Keep expanding the list until all your proxy systems are displayed.

7. Manage duplicate system names.

This step appears in the wizard if the Identity Provisioning service has already been enabled for the selected Identity Authentication tenant, and on this tenant, there are provisioning systems with the same names as the ones you selected on step 4,5 and 6.

You must provide new names for the duplicate systems.

8. Review your configurations.

On this step, you can go back and change the configurations you've made on the previous steps. You can change the target Identity Authentication tenant to which you want to migrate your Identity Provisioning tenant, revise the provisioning systems you've selected and the names you provided for the duplicate system names.

#### ⚠ Caution

Double check the data you want to migrate before you choose *Finish* on the next step. Once your migration completes successfully, you cannot trigger it again. Any data that you haven't selected for migration must be exported and manually imported into your tenant on SAP Cloud Identity infrastructure within 30 days after the successful migration.

9. Choose *Finish*.

This triggers the migration process. It cannot be stopped and cannot be reverted.

## Results

### Completed successfully

If migration completed successfully, the *Tenant Migration* tile in the Identity Provisioning admin console displays the message: Tenant already migrated. In addition, on every screen you are notified of the following:

This tenant is already migrated to `<ias-host>/admin` on SAP Cloud Identity infrastructure and will be deleted on `<date>`.

- Your Neo tenant is enabled again.
- Your migrated tenant is also enabled, however, the provisioning systems are disabled. The scheduled jobs are paused. Modified transformations are migrated with status initial, which means that you cannot reset them to an earlier version.
- Connectivity destinations are migrated as system properties. The only exception is SAP AS ABAP destination which needs to be created manually after the migration.
- A technical user called *PROXY* is created in Identity Authentication for every migrated proxy system and source system that is set up for real-time provisioning **only if** those systems have been configured with inbound certificates in the Neo tenant.

If those systems have been configured with authentication method other than inbound certificates (for example, OAuth or outgoing certificates), after migration, you need to create and configure the technical user (admin user of type System) manually in Identity Authentication. For more information, see [Add System as Administrator](#)

Proceed with the post-migration tasks in the *Next Steps* section.

### Finished with errors

If migration finished with errors, you need to see the logs for details. You are notified what has been migrated successfully and what has failed.

- Your Neo tenant is enabled. You must start over with the migration. The next time you initiate it, only systems that failed to be migrated and systems that have not been previously selected for migration will be displayed in the wizard steps.

## Next Steps

### → Recommendation

Start using your Identity Provisioning tenant on SAP Cloud Identity infrastructure. Although your Neo tenant will be available for 30 days following a successful migration, we recommend that you do not perform any operations on it, such as running jobs, adding provisioning systems and others.

If you continue working in your Neo tenant, for example: run jobs and provision users, they will be created or updated in the target systems. However, one possible implication is that you won't be able to delete the created users when you start using the tenant on SAP Cloud Identity infrastructure.

1. Log on to the target Identity Authentication tenant with your admin user.  
The URL follows the pattern: `https://<ias-host>/admin`.

### i Note

If your tenant URL is `https://<ias-host>/ips`, you are redirected to `https://<ias-host>/admin`. This opens the common administration console of SAP Cloud Identity Services, where the provisioning functionality is embedded under *Identity Provisioning* section.

2. Navigate to ► *Users & Authorizations* ► *Administrators* ►, select your admin user and assign it the *Manage Identity Provisioning* role.
3. Under *Identity Provisioning*, review your migrated provisioning systems and job schedules. Some of the provisioning systems require post-migration adjustments.
  - Proxy systems  
Update the Identity Provisioning URLs in the external application with the URLs pointing to proxy systems in your migrated Identity Provisioning tenant on SAP Cloud Identity infrastructure.

### ❖ Example

If you have configured a proxy system for provisioning user data to and from the on-premise SAP Identity Management, you need to update the value of the SCIM\_HOST repository constant in SAP Identity Management Admin UI to point to the `<ias-tenant-host>` of your migrated Identity Provisioning tenant.

### i Note

Be aware that the OAuth authentication type is now changed to Basic, therefore providing an OAuth URL for obtaining a token is no longer needed. The OAuth client ID and OAuth client secret must be replaced with the credentials of the technical user of type System created in Identity Authentication. For more information, see [3225329](#) 📄

- Real-time provisioning systems  
Update the Identity Provisioning SCIM URLs in the systems from where you want to sync the users real time. The SCIM URLs should point to the source systems (configured for real-time provisioning) in your migrated Identity Provisioning tenant on SAP Cloud Identity infrastructure.

### ❖ Example

If you have configured Identity Authentication for real-time provisioning to target systems in Identity Provisioning, in SAP Cloud Identity Services admin console, you need to update the value

of the SCIM URL field, as described in [Real-Time Provisioning in SAP Cloud Identity Infrastructure \[page 1557\]](#) ► [section III](#) ► [step 3](#) ►.

#### i Note

If you have SAP SuccessFactors Learning with Native Login authentication migrated to SAP Cloud Identity Services, the corresponding configuration on the learning solution side needs to be updated with the new URL of the SAP SuccessFactors Learning source system in the migrated Identity Provisioning tenant. For that purpose, you have to open an incident on LOD-SF-LMS-IAS component and provide the host of the migrated tenant and the ID of the SAP SuccessFactors Learning source system there.

- On-premise systems

Adjust the connection to on-premise systems. For more information, see [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#).

#### i Note

For SAP AS ABAP, you must create a connectivity destination.

4. If everything is correct with the migrated systems, go back to your Identity Provisioning tenant on Neo environment and disable the provisioning systems there.
5. Return to your migrated tenant and enable the provisioning systems.  
If a source system that is connected to a target system has not been selected for migration, a warning message will be displayed on the [Details](#) tab of the migrated target system saying that previously selected source system is invalid, deleted or missing. You'll need to select a migrated source system. Otherwise, this target system will read entities from all enabled source systems.
6. Resume your provisioning jobs.

#### i Note

The first provisioning job runs in [Full Read](#) mode, even if [Delta Read](#) has been configured. After successful full read, jobs with `ips.delta.read` set to **enabled** run as expected, that is, only modified data is provisioned from source to target systems.

## Related Information

**Blog Post:** [Go for your quick win! Migrate Identity Provisioning tenants to SAP Cloud Identity infrastructure.](#) 

## 1.7.19 Reset Identity Provisioning Tenant

Resetting your Identity Provisioning tenant deletes all systems you have set up for this tenant (subaccount), along with the relevant job execution logs.

### Context


Be careful with this option. If you reset your Identity Provisioning tenant, you will lose all systems, configurations, scheduled jobs, source system subscriptions, and all job execution logs. If you want to use the service again afterward, you will have to set up new systems.

#### → Tip

If you have a bundle account, all generated OAuth client credentials, as well as the admin users authorized for your Identity Provisioning system will be **kept**.

If want to reset your tenant, proceed as follows.

### Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Depending on the infrastructure of your Identity Provisioning tenant, follow the steps below:
  - **SAP Cloud Identity infrastructure:** Under Identity Provisioning, select the *Tenant Reset* tile
  - **Neo environment:** From the left-side menu, choose the  *Support* section and click the *Tenant Reset* link.

## 1.7.20 Reset Identity Provisioning System

Resetting an Identity Provisioning system (source or target) deletes all Identity Provisioning operational data.

### Context

There might be times when you would like to delete the current Identity Provisioning operational data for a particular system. For example, clearing entities that were read from the source system and were then mapped to SCIM specific attributes via the intermediate transformation logic.

This operation is called [system reset](#). If you choose it, you only clear the Identity Provisioning operational data. The system configurations and all existing read and provisioned entities, along with their authorizations, will be preserved. To learn more, see: [Transformations \[page 323\]](#)

If you want to reset your system, proceed as follows:

## Procedure

1. Access the Identity Provisioning UI.
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
2. Select the relevant source or target system.

### ! Restriction

This [reset](#) operation is not applicable to proxy systems.

3. Choose [Edit](#) from the top of the systems panel.
4. From the options below, choose [Reset](#).
5. Confirm with [OK](#).

## Next Steps

Regardless of the type of system you have reset - source or target ones, continue with the following steps:

1. Start a provisioning job. See: [Start and Stop Provisioning Jobs \[page 1524\]](#)
2. Set the `ips.delete.existedbefore.entities` to **true** on all affected target systems. This ensures that, if from now on you delete entities in the source system, those entities will be recognized as previously existed entities in the target systems and will be deleted there.
3. Start a provisioning job again.

### i Note

Following a reset, scheduled jobs preserve their defined time period.

## 1.8 Security

Learn more about the security features supported by the Identity Provisioning service, such as encryption rules, session management, job logs management, and customer data isolation and storage.

### Before You Start

The information below applies to both standalone and bundle tenants. To learn more about the tenant types, see: [Initial Setup of Bundle Tenants \[page 406\]](#)

### Authentication and Roles

See: [Authentication and Roles \[page 1547\]](#)

### Communication Channels

See: [Communication Security \[page 1545\]](#)

### Managing Customer Data

See: [Customer Data \[page 1545\]](#)

### Managing Logs

See: [Job Logs \[page 1548\]](#)

### Encryption

When configuring a system, always set credentials (such as passwords, private keys and OAuth secrets) as [Credential](#) properties. When you add a credential property, its value is displayed as an encrypted string. For better security, the encrypted string is always displayed as 40 characters, no matter how long your real password is.



## Session Management

Manage timeout sessions when your Identity Provisioning tenant is running on the infrastructure of SAP Cloud Identity Services or SAP BTP, Neo environment.

- **SAP Cloud Identity infrastructure** - Identity Provisioning uses the session management principles of the common infrastructure the service shares with Identity Authentication. For more information, see [Configure Session Timeout](#)
- **SAP BTP, Neo environment** - Identity Provisioning uses the session management principles of SAP Business Technology Platform. Also, no session cookies are generated. For more information, see [Authentication](#) → *Handling Session Timeout*

## Related Information

[Data Protection and Privacy \[page 1548\]](#)

### 1.8.1 Communication Security

By default, the Identity Provisioning service uses secure communication channels. Still, when connecting to customer systems, you decide (define) what the communication channel to be.

## Recommendations

- Always use secure protocols when specifying your connection details (in the cockpit → [Destinations](#) section, in the Identity Provisioning UI → [Properties](#) tab).
- Always re-enter the values of the configured credential properties when you update the URL or host name of an existing provisioning system in the [Properties](#) tab of the Identity Provisioning UI. The only exception to this are the credential properties of systems that are created with a connectivity destination.
- Avoid using property `TrustAll` in productive scenarios. When it's set to **true**, the SSL server certificate is not verified, and thus the server is not authenticated.

### 1.8.2 Customer Data

## Data Isolation

After you subscribe to the [ips](#) application, a new dedicated database schema is created for you. This guarantees that your provisioned data is stored separately, which means it's isolated from other productive customer data.

### i Note

Even if you have more than one account on one and the same landscape, you receive only one Identity Provisioning DB schema.

## Data Storage Security

In the Identity Provisioning service, no personal or sensitive information about the provisioned entities is saved. To check whether any changes have been made to an entity after the initial provisioning, the Identity Provisioning uses strong hashed algorithm for the provisioned entities.

If a provisioning job repeatedly fails and you need problem investigation, you can enable detailed entity tracing. That means, the Identity Provisioning service will log the complete information (general, personal and sensitive data) of your provisioned entities. For example, if *"groups": XXX* are not an array but have a string value instead; or an attribute that must have a string value, has an integer value instead. Tracing such data could help you identify potential incorrect attribute values for a certain entity, which you can correct in the source system or via the transformation functions.

To learn what personal and sensitive data is, see: [Glossary for Data Protection and Privacy \[page 1549\]](#)

If you want to activate entity tracing, perform as follows:

1. In your source system, set property **ips.trace.failed.entity.content** to *true*.
2. Run again the provisioning job.
3. Open the [Job Logs](#) section, select your job, and under [Failed Entities](#), choose an entity and find the log information about it.
4. If you cannot resolve the problem yourself, contact the Identity Provisioning operators. For more information, see [Getting Support \[page 1620\]](#).

### i Note

The operators may need the full trace content, so they can ask you to set the property in your target system as well, and once again run the provisioning job.

## Reset Customer Data

If you need to clear all you customer data (systems, jobs, execution logs), proceed with the steps below depending on the infrastructure of your Identity Provisioning tenant:

- **SAP Cloud Identity infrastructure:** From the [Identity Provisioning](#) section of the SAP Cloud Identity Services administration console, choose [Tenant Reset](#).

- **Neo environment:** From the [Support](#) section in the Identity Provisioning admin console, choose [Reset](#).

## Related Information

[Data Protection and Privacy \[page 1548\]](#)

## 1.8.3 Authentication and Roles

The Identity Provisioning service can be consumed either directly through its APIs, or by the user interface (UI). To operate with the service, you need to have admin permissions for the relevant Platform or bundle subaccount.

## Protection Categories

- The APIs are protected with OAuth2.0. To call an API, you need to obtain an OAuth token. See: [Register an OAuth Client](#)
- The user interface is protected with SAML2.0 authentication against the trusted identity provider configured for SAP Business Technology Platform.

### i Note

Use the service UI for provisioning entities between standard source and target systems.

Use APIs only when user interface is not available (for proxy scenarios). See: [Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

## Roles

You can provide additional users with admin rights for your consumer subaccount. Depending on the type of your tenant, you can do this either in the platform cockpit, or directly in the Identity Provisioning user interface. To learn more, see:

- [Manage Authorizations in Neo Environment \[page 1490\]](#)
- [Manage Authorizations \[page 1487\]](#)

## Related Information

[SAP Business Technology Platform: Managing Roles](#)

## 1.8.4 Job Logs

### Execution

Job logs show important information about the state of your jobs. If a job is unsuccessful, the logs will display how many entities have failed and the first few of them.

### Cleanup

Job logs are automatically deleted on a defined retention period. You can set this period to be 7, 14 or 30 days. By default, logs are kept for **7** days.

### Download

If you need to keep your job logs longer than the retention period, or just need to have them available offline, download them to your local system.

#### i Note

Logs can contain any customer data depending on what kind of information is provisioned (general or private). The Identity Provisioning service is not responsible for the content of the provisioned data. You, as administrator, can control this by the transformation logic of the systems.

### Related Information

[Monitor Provisioning Job Logs \[page 1594\]](#)

[Audit Logs: Retention and Retrieval APIs \[page 1551\]](#)

## 1.8.5 Data Protection and Privacy

Governments place legal requirements on industry to protect data and privacy. We provide features and functions to help you meet these requirements.

#### i Note

SAP does not provide legal advice in any form. SAP software supports data protection compliance by providing security features and data protection-relevant functions, such as blocking and deletion of

personal data. In some cases, compliance with applicable data protection and privacy laws may not be completely covered by the Identity Provisioning service. That's because Identity Provisioning scenarios require actions from you too, which the service cannot do for you.

Furthermore, this information should not be taken as an advice or a recommendation regarding additional features that would be required in specific IT environments. Decisions related to data protection must be made on a case-by-case basis, taking into consideration the given system landscape and the applicable legal requirements. Definitions and other terms used in this documentation are not taken from a specific legal source.

Handle personal data with care. As a data controller, you are legally responsible when processing personal data.

<a href="#">Glossary for Data Protection and Privacy [page 1549]</a>	The terms listed in this page are general to SAP products. Not all terms may be relevant for the SAP Cloud Identity Services – Identity Provisioning.
<a href="#">Audit Logs: Retention and Retrieval APIs [page 1551]</a>	<i>Change logging</i> guarantees that changes made to personal data are recorded. <i>Read-access logging</i> records access to sensitive personal data. You may be required to gather this information for auditing purposes or legal requirements.
<a href="#">Information Report [page 1552]</a>	Currently, this functionality is not applicable for the Identity Provisioning service.
<a href="#">Erasure [page 1553]</a>	When handling personal data, consider the legislation in the different countries where your organization operates. After the data has passed the end of purpose, regulations may require you to delete the data. However, additional regulations may require you to keep the data longer. During this period you must block access to the data by unauthorized persons until the end of the retention period, when the data is finally deleted.
<a href="#">Consent [page 1554]</a>	We assume that software operators, such as SAP customers, collect and store the consent of data subjects, before collecting personal data from data subjects. A data privacy specialist can later determine whether data subjects have granted, withdrawn, or denied consent.

### 1.8.5.1 Glossary for Data Protection and Privacy

The following terms are general to SAP products. Not all terms may be relevant for SAP Cloud Identity Services – Identity Provisioning.

Term	Definition
<b>Blocking</b>	A method of restricting access to data for which the primary business purpose has ended.

Term	Definition
<b>Business purpose</b>	A legal, contractual, or in other form justified reason for the processing of personal data. The assumption is that any purpose has an end that is usually already defined when the purpose starts.
<b>Consent</b>	The action of the data subject confirming that the usage of his or her personal data shall be allowed for a given purpose. A consent functionality allows the storage of a consent record in relation to a specific purpose and shows if a data subject has granted, withdrawn, or denied consent.
<b>Deletion</b>	Deletion of <b>personal data</b> so that the data is no longer available.
<b>End of business</b>	Date where the business with a data subject ends, for example the order is completed, the subscription is canceled, or the last bill is settled.
<b>End of purpose (EoP)</b>	End of purpose and start of blocking period. The point in time, when the primary processing purpose ends (for example contract is fulfilled).
<b>End of purpose (EoP) check</b>	A method of identifying the point in time for a data set when the processing of <b>personal data</b> is no longer required for the primary <b>business purpose</b> . After the <b>EoP</b> has been reached, the data is <b>blocked</b> and can only be accessed by users with special authorization (for example, tax auditors).
<b>Purpose</b>	The information that specifies the reason and the goal for the processing of a specific set of personal data. As a rule, the purpose references the relevant legal basis for the processing of personal data.
<b>Residence period</b>	The period of time between the end of business and the end of purpose (EoP) for a data set during which the data remains in the database and can be used in case of subsequent processes related to the original purpose. At the end of the longest configured residence period, the data is blocked or deleted. The residence period is part of the overall retention period.
<b>Retention period</b>	The period of time between the end of the last business activity involving a specific object (for example, a business partner) and the deletion of the corresponding data, subject to applicable laws. The retention period is a combination of the residence period and the blocking period.
<b>Personal data</b>	Any information relating to an identified or identifiable natural person ("data subject"). An identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural, or social identity of that natural person

Term	Definition
<b>Sensitive data</b>	<p>Sensitive data includes:</p> <ul style="list-style-type: none"> <li>• <a href="#">Authentication data</a> – passwords, pass phrases, certificates, tokens, and other credentials</li> <li>• <a href="#">Security-critical data</a> – cryptographic keys (except public keys), session identifiers, security configuration settings</li> <li>• <a href="#">Confidential business data</a> – all business data declared as confidential, such as financial results, sales figures, intellectual property, and other information which is useful for competitors or where unintended disclosure could harm a company</li> <li>• <a href="#">Privacy personal data</a> – see the table row above</li> </ul>
<b>Sensitive personal data</b>	<p>A category of personal data that usually includes the following type of information:</p> <ul style="list-style-type: none"> <li>• Special categories of personal data, such as data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, trade union membership, genetic data, biometric data, data concerning health or sex life or sexual orientation.</li> <li>• Personal data subject to professional secrecy</li> <li>• Personal data relating to criminal or administrative offenses</li> <li>• Personal data concerning insurances and bank or credit card accounts</li> </ul>
<b>Where-used check (WUC)</b>	<p>A process designed to ensure data integrity in the case of potential blocking of business partner data. An application's where-used check (WUC) determines if there is any dependent data for a certain business partner in the database. If dependent data exists, this means the data is still required for business activities. Therefore, the blocking of business partners referenced in the data is prevented.</p>

## 1.8.5.2 Audit Logs: Retention and Retrieval APIs

### Prerequisites

- **Standalone accounts:**  
To call the API methods, you have to create a Platform API OAuth client and obtain an access token. See: [Using Platform APIs](#)
- **Bundle accounts:**  
To view the audit logs, you have to first generate Client ID and Client Secret in the Identity Provisioning user interface. See: [Access Audit Logs \[page 1607\]](#)

## Retention API

The audit log retention API allows you to view your currently active retention period for all the audit log data that is stored for your productive Identity Provisioning subaccount. Using this API, you can also modify the default retention period (**201** days) so as to correspond to your legal, business, or other restrictions.

To learn more, see: [Audit Log Retention API Usage for the Neo Environment](#)

### ! Restriction

For bundle tenants, you cannot modify the retention period since your OAuth client is generated with the [Read Audit Logs](#) scope only.

## Retrieval API

The audit log retrieval API allows you to retrieve the audit logs for your productive Identity Provisioning subaccount. The API follows the OData 4.0 standard, providing the audit log results as OData with collection of JSON entities.

The predefined audit log message categories are:

- [audit.data-access](#) – read-access logging records for access to sensitive personal data
- [audit.data-modification](#) – data modification logging records for sensitive personal data
- [audit.security-events](#) – logging of general security events, such as login, logout, and others
- [audit.configuration](#) – logging of security critical configuration changes

To learn more, see: [Audit Log Retrieval API Usage for the Neo Environment](#)

## API Protection

Both the retention and retrieval APIs are protected with OAuth 2.0 client credentials. They provide two OAuth scopes:

- [Read Audit Logs](#) – allows usage of the Audit Log Retrieval API to retrieve audit logs
- [Manage Audit Logs](#) – allows usage of the Audit Log Retention API to read currently active retention and set custom retention.

## 1.8.5.3 Information Report

### i Note

Currently, this functionality is not applicable for the Identity Provisioning service.



The Identity Provisioning service only transfers entities from a source system to a target one. The applications representing these source and target systems may provide data about the provisioned entities but this data is only stored in the systems, not in the Identity Provisioning service itself.

## 1.8.5.4 Erasure

When handling personal data, consider the legislation in the different countries where your organization operates. After the data has passed the end of purpose, regulations may require you to delete the data. However, additional regulations may require you to keep the data longer. During this period you must block access to the data by unauthorized persons until the end of the retention period, when the data is finally deleted.

Personal data can also include referenced data. The challenge for deletion and blocking is first to handle referenced data and then other data, such as business partner data.

### i Note

If your data is stored outside SAP Business Technology Platform, we cannot guarantee that your data does not get reintegrated if you are pushing such data to our systems. You are responsible for terminating such integrations.

We cannot restore data you have in your local system.

## Account Expiration

Productive accounts expire based on the terms of your contract.

When your accounts expire, we delete your data barring legal requirements that SAP retains your data. If your organization has separate retention requirements, you are responsible for saving this data before we terminate your account.

## Disaster Recovery and Data Restore

The service maintains backups of lost data in the event of a disaster. The Identity Provisioning service uses the disaster recovery principles of SAP Business Technology Platform.

When your account is deleted, we may have this data in our backup system for the length of our backup cycle.

## Related Information

[Account Termination](#)

## 1.8.5.5 Consent

We assume that software operators, such as SAP customers, collect and store the consent of data subjects, before collecting personal data from data subjects. A data privacy specialist can later determine whether data subjects have granted, withdrawn, or denied consent.

To help you manage the consent of data subjects, the Identity Provisioning service relies on SAP Cloud Identity Service – Identity Authentication, which manages privacy policies and terms of use agreements.

For more information, see the Identity Authentication documentation: [Configuring Privacy Policies](#) and [Configuring Terms of Use](#)

## 1.9 Specific Scenarios

### Related Information

[Real-Time Provisioning \[page 1554\]](#)

[Hybrid Scenario: SAP Identity Management \[page 1565\]](#)

[Identity Directory \[page 1567\]](#)

### 1.9.1 Real-Time Provisioning

You can immediately provision entities from source to target systems.

Real-time provisioning allows you to synchronize newly created or updated users and groups from source to target systems without running manual or scheduled jobs in Identity Provisioning. This feature comes in handy for scenarios requiring synchronous provisioning, like user self-registration that needs immediate system access.

## Standard vs Real-Time

Provisioning Mode	Use Case	Key Differences	Source Systems
<i>Standard</i>	Use it for initial, regular and scheduled <a href="#">Read</a> and <a href="#">Resync</a> provisioning jobs of users and groups from any supported source system to any target system.	<ul style="list-style-type: none"> <li>Starting a provisioning job in Identity Provisioning is required.</li> <li>Users and groups (roles) are provisioned.</li> <li>Filtering properties are considered.</li> </ul>	<b>All source systems</b> (supported by Identity Provisioning)
<i>Real-Time</i>	Use it for instant provisioning of a single or a number of users or groups that were newly created or updated in the supported source system to any target system.	<ul style="list-style-type: none"> <li>You don't start a provisioning job. Changes in the user account (create, update, delete) trigger real-time sync.</li> <li>Users and groups are provisioned real time.</li> <li>Filtering properties are not considered.</li> </ul>	<ul style="list-style-type: none"> <li>Identity Authentication</li> <li>SAP SuccessFactors</li> <li>SAP SuccessFactors Learning</li> </ul>

### i Note

Real-time provisioning of groups can be configured for each source system that supports the execution of requests to the `/Groups` endpoint of the Real-time provisioning API.

## Real-Time Workflow

Configuring real-time provisioning involves the following steps:

- Create a technical user for accessing the real-time provisioning API.
  - If your Identity Provisioning is running on SAP Cloud Identity Services infrastructure, create an administrator user of type [System](#) in SAP Cloud Identity Services admin console, configure authentication and assign this user the [Access Real-Time Provisioning API](#) permission. For more information, see [Real-Time Provisioning in SAP Cloud Identity Infrastructure \[page 1557\]](#).
  - If your Identity Provisioning is running on SAP BTP, Neo environment, create OAuth client credentials in SAP BTP cockpit and assign the `IPS_PROXY_USER` role to the OAuth client. For more information, see [Real-Time Provisioning in Neo Environment \[page 1560\]](#).
- Configure the application for which you want to enable real-time provisioning as a source system in Identity Provisioning, for example SAP SuccessFactors.
- Connect this source system to one or more target systems, for example Identity Authentication.
- Enable real-time provisioning for the application that you've configured as a source system in Identity Provisioning, that is SAP SuccessFactors.  
The way you enable it varies from one application to another. However, you are always required to provide the real-time provisioning endpoint URL, the authentication type and credentials.

5. Create or update a user in the application, for example create a user with [Onboarder](#) user account type in SAP SuccessFactors.

As a result, the onboarder user will be immediately provisioned to Identity Authentication.

The following specifics are valid only when real-time provisioning is enabled for Identity Authentication:

Real-time provisioning is applicable for users that have been created or updated in SAP Cloud Identity Services admin console manually, by using SCIM API (version 1 or 2) or by upload from CSV file.

### Caution

Users coming from a source (system A) which are created or updated in Identity Authentication target (system B) using a provisioning job, **cannot** be later synchronized from Identity Authentication to another target (system C) using real-time provisioning. This is a precaution behavior, preventing any data collisions if you later decide to run a provisioning job for the same systems: Identity Authentication source (system B) and target (system C). See the example below.

### Example

1. You enable real-time provisioning for Identity Authentication and configure it as a source system that points to SAP Marketing Cloud target system.  
As a result, Identity Authentication users are immediately provisioned to SAP Marketing Cloud.
2. You configure Microsoft Active Directory as a source system that points to Identity Authentication target system and run a provisioning job.  
As a result, Microsoft Active Directory users are provisioned to Identity Authentication. Even though new users are created in Identity Authentication, they are not provisioned further to SAP Marketing Cloud by real-time provisioning.

## Supported Systems

- **Source:** Real-time provisioning scenario supports the following source systems:
  - Identity Authentication  
For more information, see [Configure Identity Provisioning Target Systems for Real-Time User Provisioning](#)
  - SAP SuccessFactors  
For more information, see [Managing Identity Authentication/Identity Provisioning Real Time Sync](#)
  - SAP SuccessFactors Learning  
For more information, see [Learning Configuration Procedure](#) → [Step 4](#)  
In bundle tenants running on SAP Cloud Identity Services infrastructure, SAP SuccessFactors Learning with Native Login authentication comes preconfigured for real-time user provisioning to Identity Authentication. Therefore, new Learning-only users will be created in Learning and then in Identity Authentication in real time.
- **Target:** Real-time provisioning scenario is applicable for all target systems supported by the Identity Provisioning service.

## Tenant Infrastructure

Configure real-time provisioning based on the infrastructure/environment your Identity Provisioning bundle or standalone tenant runs on.

- Tenants running on **SAP Cloud Identity Services infrastructure**: [Real-Time Provisioning in SAP Cloud Identity Infrastructure \[page 1557\]](#)
- Tenants running on **SAP BTP, Neo environment**: [Real-Time Provisioning in Neo Environment \[page 1560\]](#)

### 1.9.1.1 Real-Time Provisioning in SAP Cloud Identity Infrastructure

Perform real-time provisioning for newly added, updated or deleted entities when your Identity Provisioning bundle or standalone tenant is running on the infrastructure of SAP Cloud Identity Services.

- Ensure your tenant is running on SAP Cloud Identity Services infrastructure. For more information, see [Tenant Model \[page 8\]](#)

## I. Create Administrator User of Type System

1. Sign in to the SAP Cloud Identity Services administration console and select [Administrators](#) under [Users & Authorizations](#).
2. Add a new administrator user of type [System](#).  
This is the technical user for configuring the real-time provisioning. For more information, see [Add System as Administrator](#)
3. Configure the authentication method for the technical user.  
In the [Configure System Authentication](#) screen, choose one of the following options:
  - [Certificate](#)  
Upload the certificate that you generated in the application for which you enable real-time provisioning.
  - [Secrets](#)  
You need a client ID and secret. The client ID is automatically generated, therefore you only provide a secret.
4. Configure the authorization for the technical user.  
In the [Configure Authorizations](#) screen, enable the [Access Real-Time Provisioning API](#) permission. For more information, see [Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#)
5. Save your changes.

## II. Add Systems in Identity Provisioning

1. In SAP Cloud Identity Services administration console, select ► [Identity Provisioning](#) ► [Source Systems](#) ►.
2. Add and configure the application for which you want to enable real-time provisioning as a source system.  
For more information, see [Source Systems \[page 452\]](#)

3. Add and configure the target systems relevant for your real-time provisioning scenarios. For more information, see [Target Systems \[page 702\]](#)

### III. Enable Real-Time in Source Applications

In the context of real-time provisioning, source application is the system that triggers the real-time sync. Currently, you can enable it for the following applications: Identity Authentication, SAP SuccessFactors and SAP SuccessFactors Learning.

#### i Note

To provision groups real time, you need to use a REST client for initiating POST and DELETE requests to Real-Time Provisioning API: `https://<tenantId>.<host>/ipsproxy/service/api/v1/systems/<system-id>/entities/group`

#### Identity Authentication

1. In SAP Cloud Identity Services administration console, select **Users & Authorizations** **Real-Time Provisioning**.
2. Add a new system in the [Target System](#) screen and provide the following information:

New System	Field	Value
<a href="#">Target Configurations</a>	<a href="#">Display Name</a>	Provide a name for the target system. It can be the same as the one you have created in the Identity Provisioning admin console.
	<a href="#">Type</a>	From the dropdown, select <a href="#">Identity Provisioning</a> .
	<a href="#">SCIM URL</a>	<p>Provide the SCIM URL in the following pattern:</p> <p><code>https://&lt;ias-tenant-host&gt;/ipsproxy/service/api/v1/systems/&lt;system-id&gt;/entities/user</code></p> <p>The <code>&lt;system-id&gt;</code> is the ID of the <a href="#">Identity Authentication</a> source system you have added in the Identity Provisioning admin console. It is displayed at the end of the system URL.</p>

New System	Field	Value
<i>Version</i>	<i>1</i>	Defines the version of Identity Authentication SCIM API.
	<i>2</i>	<p><i>1</i> - the Identity Authentication SCIM API (in short, SCIM API version 1) is used.</p> <p><i>2</i> - the Identity Directory SCIM API (in short, SCIM API version 2) is used.</p>
<div> <div>i Note</div> <p>The version that you specify here must be the same as the version of the <code>ias.api.version</code> property in Identity Authentication source system.</p> </div>		
<i>Authentication Mechanism</i>	<i>OAuth</i>	Applicable only for customers with tenants running on Neo environment.
	<i>Basic</i>	<p>If you choose this option, provide the following information:</p> <p>In the <i>Username</i> field, enter the client ID of the technical user that was generated in the first procedure, step 3.</p> <p>In the <i>Password</i> field, enter the client secret of the technical user that was generated in the first procedure, step 3.</p>
	<i>Certificate</i>	If you choose this option, provide a common name and password and generate the certificate. Then, import it in the <i>Configure System Authentication</i> screen of the technical user for real-time provisioning, as described in the first procedure, step 3.

3. Save your changes.
4. Choose *Test Connection* before executing the provisioning. If the test is successful, you get the following message: *Connection to the selected target system was established successfully.*
5. Choose *Provision*.

For more information, see [Configure Identity Provisioning Target Systems for Real-Time User Provisioning](#)

## SAP SuccessFactors

- [Manage Real-Time Sync of New Hires from SAP SuccessFactors to Identity Authentication with Identity Provisioning](#)

- [Managing Identity Authentication/Identity Provisioning Real Time Sync](#)

## SAP SuccessFactors Learning

- [Learning Configuration Procedure](#)

### ! Restriction

Real-time provisioning of **groups** is not supported for this system.

## Result

Newly created and updated users and groups get provisioned real-time to the target systems you have configured in the Identity Provisioning UI.

## Related Information

[Configure Identity Provisioning Target Systems for User Provisioning](#)

### 1.9.1.2 Real-Time Provisioning in Neo Environment

Perform real-time provisioning for newly added, updated or deleted entities when your Identity Provisioning bundle or standalone tenant is running on the SAP BTP, Neo environment.

- Ensure your tenant is running on SAP BTP, Neo environment. For more information, see [Tenant Model \[page 8\]](#)

#### ⚠ Caution

Effective September 2020, Neo tenants from Shanghai (China) should be accessed on the following domain: `dispatcher.cn1.platform.sapcloud.cn`

For example: <https://ips-abc1234567.dispatcher.cn1.platform.sapcloud.cn>

So make sure you use the correct domain as `<host>` when you construct your OAuth URL (Procedure I, step 6) and SCIM URL (Procedure II, step 4).

## I. Create OAuth Client Credentials in SAP BTP Cockpit

### i Note

Step 1 and 2 are relevant for bundle tenants. If you are using standalone tenants, start from step 3.



1. Log on to the *Identity Provisioning* admin console and navigate to ► **Security** ► **Authorizations** ►  
For more information, see [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#).
2. In the *Configure Authorizations* screen, enable *Manage OAuth Clients* permission for the admin user who will configure real-time provisioning.  
For more information, see [Manage Authorizations \[page 1487\]](#)
3. Log on to the SAP BTP cockpit → *Neo* → *Overview*.  
The URL follows the pattern: `https://account.<neo_region>.hana.ondemand.com/cockpit`.  
For more information, see [Regions and Hosts Available for the Neo Environment](#)

### Note

You can see the Global provider account, which SAP provides for your bundle in the corresponding Identity Provisioning region. In the global account, you can see your subaccount, where the Identity Provisioning is enabled for your bundle. The display name of the subaccount starts with **SAP\_BUNDLE**.

4. Register a new OAuth client for the subscription to the *ipsproxy* application.
  1. Navigate to ► **Security** ► **OAuth** ► **Clients** ►.
  2. Choose *Register New Client*.
  3. From the *Subscription* combo box, select **<provider\_subaccount>/ipsproxy**.
  4. From the *Authorization Grant*, select **Client Credentials**.
  5. Save (in a notepad) the generated *Client ID* or rename it for your convenience. You will need it for the real-time configuration steps.
  6. In the *Secret* field, enter a password (client secret) and remember it.
5. Navigate to ► **Security** ► **OAuth** ► **Branding** ►.
6. From the *OAuth URLs* section, copy and save (in a notepad) the **Token Endpoint** URL. You will need it for the real-time configuration steps, too.  
It follows the pattern: `https://oauthservices-<consumer_subaccount>.<host>/oauth2/api/v1/token`  
For example: `https://oauthservices-xyz12345.hana.ondemand.com/oauth2/api/v1/token`
7. Assign the **IPS\_PROXY\_USER** role to the OAuth client:
  1. On your subaccount level, navigate to ► **Subscriptions** ► **Subscribed Java Applications** ► and choose *ipsproxy*.
  2. From the left-side navigation, choose *Roles*.
  3. Assign the **IPS\_PROXY\_USER** role to the newly created OAuth client. Choose *Assign* and enter the `oauth_client-<client_ID>`, where **<client\_ID>** is the ID you have saved on step 4.e.

## II. Add Systems in Identity Provisioning

1. Log on to the *Identity Provisioning* admin console and select the *Source Systems* tile.
2. Add and configure the application for which you want to enable real-time provisioning as a source system.  
For more information, see [Source Systems \[page 452\]](#)
3. Add and configure the target systems relevant for your real-time provisioning scenarios. For more information, see [Target Systems \[page 702\]](#)
4. Construct and save (in a notepad) the SCIM URL. You'll need it for the real-time configuration steps. It follows the pattern:

- **Basic Authentication:**

For user: `https://ipsproxy<provider_account>-<consumer_subaccount>.<host>/ipsproxy/api/v1/systems/Identity_Authentication_ID/entities/user`

- **Certificate Authentication:**

For user: `https://ipsproxy<provider_account>-<consumer_subaccount>.cert.<host>/ipsproxy/certapi/v1/systems/Identity_Authentication_ID/entities/user`

→ Tip

- `<provider_account>` is the Global provider account from **step 3** in the first procedure.
- `<Identity_Authentication_ID>` is the ID of the *Identity Authentication* source system, displayed at the end of the system URL in the Identity Provisioning admin console.

To get the correct application URLs, navigate to ► *Subscriptions* ► *Subscribed Java Applications* ► *ipsproxy* ► *Application URLs* ►.

### III. Enable Real-Time in Source Applications

In the context of real-time provisioning, source application is the system that triggers the real-time sync. Currently, you can enable it for the following applications: Identity Authentication, SAP SuccessFactors and SAP SuccessFactors Learning.

#### i Note

To provision groups real time, you need to use a REST client for initiating POST and DELETE requests to Real-Time Provisioning API: `https://ipsproxy<provider_account>-<consumer_subaccount>.<host>/ipsproxy/api/v1/systems/<system-id>/entities/group`

#### Identity Authentication

1. Sign in to the SAP Cloud Identity Services administration console.  
The URL follows the pattern: `https://<ias-host>/admin`

#### ⚠ Caution

For Shanghai (China) tenants, the URL pattern is: `https://<tenant_ID>.accounts.sapcloud.cn/admin`

2. Navigate to ► *Users & Authorizations* ► *Real-Time Provisioning* ►.
3. Add a new system in the *Target System* screen and provide the following information:

New System	Field	Value
<i>Target Configurations</i>	<i>Display Name</i>	Provide a name for the target system. It can be the same as the one you have created in the Identity Provisioning admin console.

New System	Field	Value
	<i>Type</i>	From the dropdown, select <b>Identity Provisioning</b> .
	<i>SCIM URL</i>	Provide the URL you have configured in the second procedure, step 4.
<i>Version</i>	<i>1</i>	Defines the version of Identity Authentication SCIM API.
	<i>2</i>	<p><i>1</i> - the Identity Authentication SCIM API (in short, SCIM API version 1) is used.</p> <p><i>2</i> - the Identity Directory SCIM API (in short, SCIM API version 2) is used.</p>
<div> <b>i Note</b> <p>The version that you specify here must be the same as the version of the <code>ias.api.version</code> property in Identity Authentication source system.</p> </div>		
<i>Authentication Mechanism</i>	<i>OAuth</i>	<p>Select <i>OAuth</i> authentication mechanism.</p> <ul style="list-style-type: none"> <li>In the <i>OAuth URL</i>, provide the token endpoint URL you have saved in the first procedure, on step 6.</li> <li>In the <i>Client ID</i> field, provide the OAuth client ID from the first procedure, step 4.e.</li> <li>In the <i>Client Secret</i> field, provide the OAuth client secret from the first procedure, step 4.f.</li> </ul>
	<i>Basic</i>	Applicable only for customers with tenants running on SAP Cloud Identity Services infrastructure.

New System	Field	Value
	<a href="#">Certificate</a>	<p>If you choose this option, provide a common name and password and generate the certificate. Then, select the Identity Authentication source system in Identity Provisioning admin console and import it in the <a href="#">Inbound Certificates</a> tab.</p> <p>For more information, see <a href="#">Manage Certificates for Inbound Connection [page 1510]</a></p>

4. Save your changes.
5. Choose [Test Connection](#) before executing the provisioning.  
If the test is successful, you get the following message: *Connection to the selected target system was established successfully.*
6. Choose [Provision](#).

For more information, see [Configure Identity Provisioning Target Systems for Real-Time User Provisioning](#)

## SAP SuccessFactors

- [Manage Real-Time Sync of New Hires from SAP SuccessFactors to Identity Authentication with Identity Provisioning](#)
- [Managing Identity Authentication/Identity Provisioning Real Time Sync](#)

## SAP SuccessFactors Learning

- [Learning Configuration Procedure](#)

### ! Restriction

Real-time provisioning of **groups** is not supported for this system.

## Result

Newly created and updated users and groups get provisioned real-time to the target systems you have configured in the Identity Provisioning UI.

## 1.9.2 Hybrid Scenario: SAP Identity Management

You can execute hybrid scenarios between connectors from the Identity Provisioning UI and external consumer systems (back-ends) that support SCIM 2.0 protocol.

### Prerequisites

- You have a productive Identity Provisioning service (standalone version or a bundle tenant).
- You have user credentials for an external system (like SAP Identity Management) with read and write permissions.

#### i Note

Administrators of bundle tenants on Neo environment should enable the [Manage OAuth Clients](#) permission, as described in *Neo Environment* section in [Manage Authorizations \[page 1487\]](#).

### Context

A proxy system is a special connector type you can use for [hybrid](#) scenarios. That means, you can provision entities from one system to another (and the other way around) without making a direct connection between them. To achieve this, you can add an Identity Provisioning proxy system, based on SCIM 2.0 protocol. It will initially provision entities to an external consumer system and then will start executing CRUD operations backwards, whenever the external system requests such.

A SCIM 2.0 system can act as a proxy if it supports both read and write operations. To check which system types are appropriate for this role, see the table in topic [Proxy Systems \[page 981\]](#).

The scenario below is exemplary. It involves a proxy connector you have added in the Identity Provisioning UI, and **SAP Identity Management** as an external consumer system.

### Procedure

1. Open your subaccount in SAP BTP cockpit.

#### i Note

If you have a bundle tenant, then in the cockpit → [Neo](#) → [Overview](#), you can see the Global account, which SAP provides for your bundle in the corresponding Identity Provisioning region. Then, in the global account, you can see your subaccount, where the Identity Provisioning is enabled as a service for the bundle. The display name of the subaccount starts with **SAP\_BUNDLE\_**.

2. Register a new OAuth client for the subscription to the [ipsproxy](#) application:
  1. Go to ► [Security](#) ► [OAuth](#) ► [Clients](#) ►.

2. Choose [Register New Client](#).
3. From the [Subscription](#) combo box, select **<provider\_subaccount>/ipsproxy**.
4. From the [Authorization Grant](#) combo box, select **Client Credentials**.
5. In the [Secret](#) field, enter a password (client secret) and remember it. You will need it later, for the repository configuration in SAP Identity Management.
6. Copy/paste and save (in a notepad) the generated [Client ID](#). You will need it later, too.
3. Assign role IPS\_PROXY\_USER to the OAuth client:
  1. From the left-side navigation, choose [Subscriptions](#).
  2. Under the [Java Applications](#) section, choose [ipsproxy](#).
  3. From the left-side navigation, choose [Roles](#).
  4. Assign role **IPS\_PROXY\_USER** to the newly created OAuth client. Choose [Assign](#) and enter **oauth\_client\_<client\_ID>**, where <client\_ID> is the one from step 2.f.
4. Now open the Identity Provisioning admin console. The access URL depends on the productive type of your Identity Provisioning. See:
  - [Access Identity Provisioning UI of Bundle Tenants \[page 413\]](#)
  - [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
5. Add a proxy system. You can choose among a list of systems available for hybrid scenarios with SAP Identity Management. To find the system you need, see [Proxy Systems \[page 981\]](#) → **More Information**.
6. Open the [Properties](#) tab of the selected proxy system to configure its connection settings.
7. If necessary, modify the default [Read](#) and [Write](#) transformation mapping rules to reflect the current setup of entities in your proxy system.
8. Save your changes.

## Next Steps

1. Now, you can export the newly created proxy system. To do that, choose [Export](#) → [CSV format](#).
2. Then, go to SAP Identity Management to register or import a SCIM repository.

### i Note

If you import the [.csv](#) file, you will have all the fields automatically filled-in. However:

- You will need to manually enter your client ID and secret (AUTH\_USER and AUTH\_PASSWORD).
- For the SCIM\_ASSIGNMENT\_METHOD constant, leave its default value: **PATCH**.  
However, check explicitly in the documentation of the relevant proxy system type whether you should leave the default value **PATCH**, or change it to **PUT**. Find your system under section [Proxy Systems \[page 981\]](#).

3. Then start an [Initial load](#) job. After the initial load is done, you can create new users or update existing ones in SAP Identity Management.

## Future Identity Lifecycle

What happens when you make new changes (create/update/delete an entity)?

- If a new change is made in SAP Identity Management, a new job is automatically triggered. It applies the changes in the external back-end system so that both systems become up-to-date and synchronized.
- If a new change is made in the back-end system, you have to start a new *Initial load* job so that the changes can apply in SAP Identity Management.

## Related Information

SAP Identity Management: [Setting Up Hybrid Integration with a Cloud System](#)

SAP Community Blog: [Hybrid Scenarios with Identity Provisioning Proxy](#)

## 1.9.3 Identity Directory

Identity Directory is the persistency layer of SAP Cloud Identity Services. It offers a central place for storing and managing users and groups. Provisioning of these entities to and from the directory is ensured by the Local Identity Directory connector of Identity Provisioning service.

## Concept

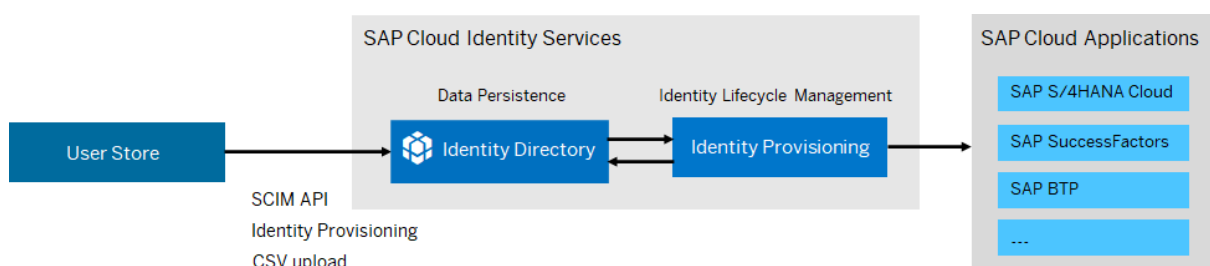
### Note

The *Local Identity Directory* connector is available for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

The identity directory provides a System for Cross-domain Identity Management (SCIM) 2.0 REST API for managing resources (users, groups and custom schemas) with a set of attributes. Those attributes are defined in the SCIM 2.0 Core schema and the Enterprise user resource schema. Custom attributes are supported through a schema extension. For more information, see [Identity Directory Service → API Reference](#).

You can use Identity Provisioning for reading entities from user stores like, Microsoft Active Directory, SAP Identity Management and others, and replicating them to the directory. From there, you can provision them further to various cloud systems, like SAP S/4HANA Cloud.

The figure below shows an example of a system landscape you can use for provisioning scenarios with the identity directory.



For every newly created user, the directory generates `Global User ID` - the unique user identifier across landscape. This identifier can be generated externally, too. Afterwards, Identity Provisioning distributes the Global User ID to SAP cloud applications, like SAP Task Center, which need the common user identifier in their integration scenarios. For more information, see [Global User ID in Integration Scenarios](#)

## Use Cases

- Classic use case - the Local Identity Directory can be configured both as a target and as a source system. See: [Configuring Local Identity Directory in Target-Source Scenario \[page 1568\]](#)
- Proxy mode - the Local Identity Directory as a proxy connector. See: [Configuring Local Identity Directory in Proxy Scenario \[page 1579\]](#)
- Merging attributes - read attributes of a single user that exists in multiple source systems and merge the attributes in the directory. See: [Patched and Merged Attributes \[page 1590\]](#)

## Related Information

**Blog Post:** [SAP Cloud Identity Services – Identity Directory](#) 

**Blog Post:** [SAP Cloud Identity Services – Why and how to integrate them for a consistent identity lifecycle?](#) 

### 1.9.3.1 Configuring Local Identity Directory in Target-Source Scenario

Configure the Local Identity Directory both as a target and as a source system in the Identity Provisioning UI.

## Context

### Note

The *Local Identity Directory* connector is available for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

In a typical use case, the Local Identity Directory is first configured as a target system, where users and groups are provisioned to, and then configured as a source system, from where users and groups are read and provisioned to target systems.

An example scenario includes the following systems: *Local Identity Directory*, *SAP SuccessFactors*, and *Microsoft Azure Active Directory*.



## Procedure

1. Sign in to SAP Cloud Identity Services administration console and navigate to ► [Identity Provisioning](#) ► [Target Systems](#) ►.
2. Add [SAP SuccessFactors](#) as a source system. For more information, see [SAP SuccessFactors \[page 635\]](#).

You can provision entities from one or from multiple source systems to a single [Local Identity Directory](#) target system. For more information about merging data from multiple source systems, see: [Patched and Merged Attributes \[page 1590\]](#)

3. Add [Local Identity Directory](#) as a target system. It's configured by default, thus you don't need to enter connectivity properties or credentials.
4. **Optional:** Modify the transformation.

**Default write transformation** of Local Identity Directory

### Code Syntax

```
{
  "user": {
    "condition": "($.emails EMPTY false) && ($.userName EMPTY false) &&
isValidEmail($.emails[0].value)",
    "mappings": [
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant": [
          "urn:ietf:params:scim:schemas:core:2.0:User",
          "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
          "urn:ietf:params:scim:schemas:extension:sap:2.0:User",
          "urn:sap:cloud:scim:schemas:extension:custom:2.0:User"
        ],
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName"
      },
      {
        "constant": "userName",
        "targetVariable": "entityCorrelationAttributeName"
      },
      {
        "sourcePath": "$.userName",
        "targetVariable": "entityCorrelationAttributeValue"
      },
      {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$"
      },
      [
        "urn:ietf:params:scim:schemas:extension:sap:2.0:User"
      ][
        'emails'
      ],
      "scope": "createEntity",
      "functions": [
        {
          "function": "putIfAbsent",
          "key": "verified",
          "defaultValue": true
        }
      ]
    ]
  },
}
```

```

"targetPath": "$['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['emails'][*]['type']",
  "type": "remove"
},
{
  "sourcePath": "$.emails[*].value",
  "preserveArrayWithSingleElement": true,
  "targetPath": "$.emails[?(@.value)]"
},
{
  "sourcePath": "$.name.givenName",
  "targetPath": "$.name.givenName",
  "optional": true
},
{
  "sourcePath": "$.name.middleName",
  "targetPath": "$.name.middleName",
  "optional": true
},
{
  "sourcePath": "$.name.familyName",
  "targetPath": "$.name.familyName",
  "optional": true
},
{
  "sourcePath": "$.name.honorificPrefix",
  "targetPath": "$.name.honorificPrefix",
  "optional": true
},
{
  "sourcePath": "$.addresses",
  "targetPath": "$.addresses",
  "preserveArrayWithSingleElement": true,
  "defaultValue": [],
  "optional": true,
  "functions": [
    {
      "function": "putIfAbsent",
      "key": "type",
      "defaultValue": "work"
    },
    {
      "condition": "(@.type NIN ['work', 'home'])",
      "function": "putIfPresent",
      "key": "type",
      "defaultValue": "work"
    }
  ]
},
{
  "sourcePath": "$.phoneNumbers",
  "targetPath": "$.phoneNumbers",
  "preserveArrayWithSingleElement": true,
  "optional": true
},
{
  "sourcePath": "$.displayName",
  "targetPath": "$.displayName",
  "optional": true
},
{
  "sourcePath": "$.userType",
  "targetPath": "$.userType",
  "optional": true
},
{
  "sourcePath": "$.locale",

```

```

        "targetPath": "$.locale",
        "optional": true
    },
    {
        "sourcePath": "$.timezone",
        "targetPath": "$.timezone",
        "optional": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "defaultValue": true,
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",

```

```

        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional" : true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['manager']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['manager']
['value']",
        "optional" : true,
        "functions": [
            {
                "function": "resolveEntityIds"
            }
        ]
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['manager']
['displayName']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User'] ['manager']
['displayName']",
        "optional" : true
    },
    {
        "constant": false,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User'] ['sendMail']",
        "scope": "createEntity"
    },
    {
        "constant": true,
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User'] ['mailVerified']",
        "scope": "createEntity"
    },
    {
        "constant": "disabled",

"targetPath": "$['urn:ietf:params:scim:schemas:extension:sap:2.0:User']
['passwordDetails'] ['status']",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User'] ['attributes']",
        "preserveArrayWithSingleElement": true,
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User'] ['attributes']"
    },
    {
        "constant": "<your-initial-password>",
        "targetPath": "$.password",
        "scope": "createEntity",
        "ignore": "true"
    },
    {
        "constant": "<your-source-system-type-code>",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User'] ['sourceSystem']",
        "scope": "createEntity",
        "ignore": true
    },

```

```

        {
          "constant": "<your-source-system-id>",
          "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId']",
          "scope": "createEntity",
          "ignore": true
        }
      ],
    },
    "group": {
      "mappings": [
        {
          "sourceVariable": "entityIdTargetSystem",
          "targetPath": "$.id"
        },
        {
          "constant":
            ["urn:ietf:params:scim:schemas:core:2.0:Group", "urn:sap:cloud:scim:schemas:
            extension:custom:2.0:Group"],
          "targetPath": "$.schemas"
        },
        {
          "sourcePath": "$.displayName",
          "targetPath": "$.displayName"
        },
        {
          "sourcePath": "$.members[*].value",
          "preserveArrayWithSingleElement": true,
          "optional": true,
          "targetPath": "$.members[?(@.value)]",
          "functions": [
            {
              "entityType": "user",
              "type": "resolveEntityIds"
            }
          ]
        },
        {
          "sourcePath": "$.members[*].value",
          "preserveArrayWithSingleElement": true,
          "optional": true,
          "targetPath": "$.members[?(@.value)]",
          "functions": [
            {
              "entityType": "group",
              "type": "resolveEntityIds"
            }
          ]
        },
        {
          "sourcePath": "$.displayName",
          "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
          "scope": "createEntity"
        },
        {
          "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
          "optional": true,
          "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
          "scope": "createEntity"
        },
        {
          "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['description']",
          "optional": true,

```

```

        "targetPath": "$
    [ 'urn:sap:cloud:scim:schemas:extension:custom:2.0:Group' ][ 'description' ]"
    }
  ]
}

```

5. **Optional:** Subscribe to the [SAP SuccessFactorsMonitor Provisioning Job Logs \[page 1594\]](#) and [Manage Job Notifications \[page 1605\]](#).
6. Start a provisioning job for the [SAP SuccessFactors](#) source system.
7. Add [Local Identity Directory](#) as a source system. It already contains all the users provisioned from the **source system to beSAP SuccessFactors** system.

It's configured by default too, and you don't need to enter any connectivity properties or credentials. However, if you want the identity directory to read only particular users or groups, open the [Properties](#) tab and enter the following properties:

Name	Value
<code>idds.group.filter</code>	<p>Filters groups by display name. You can set a single display name or multiple ones as filter criteria. If you enter multiple display names (using OR operator), the filter will search for any of them.</p> <p>For example:</p> <ul style="list-style-type: none"> <li><code>displayName eq "FellowshipTeam1"</code></li> <li><code>displayName eq "FellowshipTeam1" or displayName eq "JuniorTest3"</code></li> </ul>

Name	Value
<code>ids.user.filter</code>	Filters users by particular attributes. You can set a single attribute or multiple ones as search criteria.

→ Tip

The following SCIM user attributes are supported for filtering: *userName*, *displayName*, *emails.value*, *roles.value*, *locality*, *region*, *postalCode*, *country*, *manager*, *employeeNumber*, *costCenter*, *organization*, *division*, *department*

For example:

- *userName eq "Sebastian"*
- Using OR: *userName eq "Sebastian" or addresses.country eq "France"*
- Using AND: *userName eq "Sebastian" and addresses.country eq "France"*
- Using brackets ( ): *userName eq "Sebastian" or (addresses.country eq "France" and emails.value eq "sebastian123@mail.com")*
- Using enterprise attributes:  
*urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:department eq "Dev" and urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:organization eq "Technology"*

## 8. Modify the transformation

**Default read transformation** of Local Identity Directory.

### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable": "entityIdSourceSystem",
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']['userId']"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['userUuid']"
      },
      {
        "sourcePath": "$.schemas",
        "preserveArrayWithSingleElement": true,
        "targetPath": "$.schemas"
      },
      {
        "sourcePath": "$.userName",
        "targetPath": "$.userName",
        "optional": true,
        "correlationAttribute": true
      }
    ]
  }
}
```

```

{
  "sourcePath": "$.name.givenName",
  "targetPath": "$.name.givenName",
  "optional": true
},
{
  "sourcePath": "$.name.middleName",
  "targetPath": "$.name.middleName",
  "optional": true
},
{
  "sourcePath": "$.name.familyName",
  "targetPath": "$.name.familyName",
  "optional": true
},
{
  "sourcePath": "$.name.honorificPrefix",
  "targetPath": "$.name.honorificPrefix",
  "optional": true
},
{
  "sourcePath": "$.emails[*].value",
  "preserveArrayWithSingleElement": true,
  "targetPath": "$.emails[?(@.value)]"
},
{
  "sourcePath": "$.emails[?(@.primary== true)].value",
  "correlationAttribute": true
},
{
  "sourcePath": "$.active",
  "targetPath": "$.active"
},
{
  "sourcePath": "$.userType",
  "targetPath": "$.userType",
  "optional": true
},
{
  "sourcePath": "$.addresses",
  "targetPath": "$.addresses",
  "preserveArrayWithSingleElement": true,
  "optional": true
},
{
  "sourcePath": "$.locale",
  "targetPath": "$.locale",
  "optional": true
},
{
  "sourcePath": "$.phoneNumbers",
  "targetPath": "$.phoneNumbers",
  "preserveArrayWithSingleElement": true,
  "optional": true
},
{
  "sourcePath": "$.timezone",
  "targetPath": "$.timezone",
  "optional": true
},
{
  "sourcePath": "$.displayName",
  "targetPath": "$.displayName",
  "optional": true
},
{
  "sourcePath": "$.groups",
  "targetPath": "$.groups",

```



```

        "preserveArrayWithSingleElement": true,
        "optional": true
      },
      {
        "type": "remove",
        "targetPath": "$.groups[*].display"
      },
      {
        "condition": "$.displayName EMPTY true",
        "type": "remove",
        "targetPath": "$.displayName"
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validFrom']",
        "optional": true
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['validTo']",
        "optional": true
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystem']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystem']",
        "optional": true
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:extension:sap:2.0:User']['sourceSystemId']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
        "optional": true
      },
      {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
      },
      {

```

```

        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['value']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['value']",
        "optional": true
    },
    {
        "sourcePath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['displayName']",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']",
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:User']",
        "optional": true
    },
    {
        "sourcePath": "$.company",
        "targetPath":
"$['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
        "optional": true
    }
],
"group": {
    "mappings": [
        {
            "sourcePath": "$.id",
            "targetVariable": "entityIdSourceSystem"
        },
        {
            "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']
['name']",
            "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']
['name']"
        },
        {
            "sourcePath": "$.displayName",
            "targetPath": "$.displayName"
        }
    ]
}

```

```

    "sourcePath": "$.members",
    "targetPath": "$.members",
    "preserveArrayWithSingleElement": true,
    "optional": true
  }
}

```

9. Add [Microsoft Azure Active Directory](#) as a target system. For more information, see [Microsoft Azure Active Directory \[page 969\]](#).
10. Start another provisioning job – for the [Local Identity Directory](#) source system. We recommend that you subscribe to receive notifications from this system, too.
11. Verify if everything is successfully provisioned.

## Related Information

[Identity Directory \[page 1567\]](#)

[Patched and Merged Attributes \[page 1590\]](#)

### 1.9.3.2 Configuring Local Identity Directory in Proxy Scenario

Configure the identity directory as a proxy system in the Identity Provisioning UI.


## Context

### Note

The [Local Identity Directory](#) connector is available for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

You can use the identity directory as a SCIM 2.0 based proxy connector. Before exposing it to an external system, you should first populate it with users and groups. To learn how to read users and groups, and then provision them to the identity directory, see: [Configuring Local Identity Directory in Target-Source Scenario \[page 1568\]](#)

### SCIM Filtering Support

The Identity Provisioning proxy application supports a limited SCIM implementation based on the [SCIM Query](#)  standard. That means, you can use 'eq' filters by one SCIM attribute, and it's only applicable to users. If your system supports [native read filtering](#), the Identity Provisioning proxy application will translate the SCIM filter to the native system filter, and will try to combine it along with the relevant read filter property, if such is present.

If the Identity Provisioning finds:

- **0** users that meet the filtering criteria, the service returns HTTP status code **200** (OK) with '*totalResults*' set to a value of 0.
- **1** user that meets the filtering criteria, the service returns HTTP status code **200** (OK), and includes the result in the body of the response.
- **More than 1** users that meet the filtering criteria, the service returns HTTP status code **400** (Bad Request) with detail error type '*tooMany*'.

Bear in mind the following restrictions:

- In the [Read Transformation](#), there must be mapping between the attribute names in "sourcePath" and "targetPath" (see the example mapping below, where targetPath matches the left side of the used '*eq*' filter).
- Fully qualified names (**<schema>:<attribute>**) are not supported. For example:  
[GET .../Users/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber eq '<attribute>'](#)
- If your system supports multivalued e-mails (that is [\\$.emails\[0\].value](#), [\\$.emails\[1\].value](#), etc.), the search criteria will always resolve only one user e-mail. For SCIM-based systems, this is the first user e-mail ([\\$.emails\[0\].value](#)).

#### ❖ Example

Extract from the user mapping in the Read Transformation:

```
{
  "sourcePath": "$.userName",
  "targetPath": "$.userName",
  "correlationAttribute": true
},
```

You also set the following filter in the [Properties](#) tab: `ids.user.filter = addresses.country eq "France"`

Then if, for example, the SCIM Proxy endpoint request is: **GET .../Users?filter=userName eq "johnsmith03"**

The query request to the SCIM system API will result into: **/Users?filter=addresses.country eq "France" and userName eq "johnsmith03"**

## Procedure

1. Sign in to SAP Cloud Identity Services administration console and create a technical user with the necessary authorizations. It will later be used by the external consumer to connect to Identity Provisioning.

If you already have a technical user, skip this step.

- For **Certificate-based authentication**, follow the procedure in [Manage Certificates for Inbound Connection \[page 1510\]](#) → *SAP Cloud Identity Infrastructure*
- For **Basic authentication**, navigate to ► [Users & Authorizations](#) ► [Administrators](#) ► and add an admin user of type **System**. Configure a client ID and secret for this user and enable the [Access Proxy System API](#) permission.

2. Navigate to ► [Identity Provisioning](#) ► [Proxy Systems](#) ►
3. Add [Local Identity Directory](#) as a proxy system.

For more information, see [Add a System \[page 1477\]](#).

It's configured by default, so you don't need to enter any properties or credentials. However, if you want the identity directory to read only particular users or groups, open the [Properties](#) tab and enter the following properties:

Name	Value
<code>ids.group.filter</code>	<p>Filters groups by display name. You can set a single display name or multiple ones as filter criteria. If you enter multiple display names (using OR operator), the filter will search for any of them.</p> <p>For example:</p> <ul style="list-style-type: none"> <li>• <code>displayName eq "FellowshipTeam1"</code></li> <li>• <code>displayName eq "FellowshipTeam1" or displayName eq "JuniorTest3"</code></li> </ul>
<code>ids.user.filter</code>	<p>Filters users by particular attributes. You can set a single attribute or multiple ones as search criteria.</p> <div style="border: 1px solid #ccc; background-color: #f9f9f9; padding: 10px; margin-top: 10px;"> <p>→ <b>Tip</b></p> <p>The following SCIM user attributes are supported for filtering: <code>userName</code>, <code>displayName</code>, <code>emails.value</code>, <code>roles.value</code>, <code>locality</code>, <code>region</code>, <code>postalCode</code>, <code>country</code>, <code>manager</code>, <code>employeeNumber</code>, <code>costCenter</code>, <code>organization</code>, <code>division</code>, <code>department</code></p> </div> <p>For example:</p> <ul style="list-style-type: none"> <li>• <code>userName eq "Sebastian"</code></li> <li>• Using OR: <code>userName eq "Sebastian" or addresses.country eq "France"</code></li> <li>• Using AND: <code>userName eq "Sebastian" and addresses.country eq "France"</code></li> <li>• Using brackets ( ): <code>userName eq "Sebastian" or (addresses.country eq "France" and emails.value eq "sebastian123@mail.com")</code></li> <li>• Using enterprise attributes:  <code>urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:department eq "Dev" and urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:organization eq "Technology"</code></li> </ul>

4. **Optional:** Modify the transformations.

Transformations are used to map the user attributes from the data model of the source system to the data model of the target system, and the other way around. The identity directory offers default read and write transformations, whose settings are displayed in the Identity Provisioning UI after creating and saving the proxy system.

You can change the default transformation mapping rules to reflect your current setup of entities in your identity directory. For more information, see [Manage Transformations \[page 1494\]](#).

**Default transformation** of Local Identity Directory

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetVariable":
"entityIdSourceSystem",
        "targetPath": "$.id"
      },
      {
        "sourceVariable":
"entityBaseLocation",
        "targetVariable":
"entityLocationSourceSystem",
        "targetPath":
"$ .meta.location",
        "functions": [
          {
            "type":
"concatString",
            "suffix": "$
{entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['userUid']",
        "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']['userUid']"
      },
      {
        "sourcePath": "$ .schemas",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$ .schemas"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "optional": true,
        "correlationAttribute":
true
      },
      {
        "sourcePath":
"$ .emails[*].value",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$ .emails[?
(@.value)]"
      }
    ]
  }
}

```

## Code Syntax

```

{
  "user": {
    "scimEntityEndpoint": "Users",
    "mappings": [
      {
        "sourceVariable":
"entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "constant":
["urn:ietf:params:scim:schemas:cor
e:2.0:User","urn:ietf:params:scim:
schemas:extension:enterprise:2.0:U
ser","urn:ietf:params:scim:schemas
:extension:sap:2.0:User"],
        "targetPath": "$ .schemas"
      },
      {
        "sourcePath":
"$ .userName",
        "targetPath":
"$ .userName",
        "optional": true
      },
      {
        "sourcePath":
"$ .emails[*].value",

"preserveArrayWithSingleElement":
true,
        "targetPath": "$ .emails[?
(@.value)]"
      },
      {
        "sourcePath":
"$ .userType",
        "targetPath":
"$ .userType",
        "optional": true
      },
      {
        "sourcePath":
"$ .name.givenName",
        "targetPath":
"$ .name.givenName",
        "optional": true
      },
      {
        "sourcePath":
"$ .name.middleName",
        "targetPath":
"$ .name.middleName",
        "optional": true
      },
      {
        "sourcePath":
"$ .name.familyName",
        "targetPath":
"$ .name.familyName",

```

```

    },
    {
      "sourcePath":
"$$.emails[0].value",
      "targetPath":
"$$.emails[0].value"
    },
    {
      "sourcePath": "$$.emails[?
(@.primary== true)].value",
      "correlationAttribute":
true
    },
    {
      "sourcePath": "$$.active",
      "targetPath": "$$.active"
    },
    {
      "sourcePath":
"$$.userType",
      "targetPath":
"$$.userType",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.givenName",
      "targetPath":
"$$.name.givenName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.middleName",
      "targetPath":
"$$.name.middleName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.familyName",
      "targetPath":
"$$.name.familyName",
      "optional": true
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "targetPath":
"$$.name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$$.addresses",
      "targetPath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },

```

```

      "optional": true
    },
    {
      "sourcePath":
"$$.name.honorificPrefix",
      "targetPath":
"$$.name.honorificPrefix",
      "optional": true
    },
    {
      "sourcePath":
"$$.addresses",
      "targetPath":
"$$.addresses",
      "preserveArrayWithSingleElement":
true,
      "defaultValue": [],
      "optional": true,
      "functions": [
        {
          "function":
"putIfAbsent",
          "key": "type",
          "defaultValue": "work"
        },
        {
          "condition": "(@.type
NIN ['work', 'home'])",
          "function":
"putIfPresent",
          "key": "type",
          "defaultValue": "work"
        }
      ]
    },
    {
      "sourcePath": "$$.locale",
      "targetPath": "$$.locale",
      "optional": true
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "targetPath":
"$$.phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName",
      "optional": true
    },
    {
      "sourcePath": "$$
[urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User]"
      "validFrom": ""
    },

```



```

    {
      "sourcePath": "$.locale",
      "targetPath": "$.locale",
      "optional": true
    },
    {
      "sourcePath":
"$$.phoneNumbers",
      "targetPath":
"$$.phoneNumbers",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath":
"$$.timezone",
      "targetPath":
"$$.timezone",
      "optional": true
    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName",
      "optional": true
    },
    {
      "sourcePath":
"$$.sourceSystem",
      "targetPath":
"$$.sourceSystem",
      "ignore": true
    },
    {
      "sourcePath": "$.groups",
      "targetPath": "$.groups",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "type": "remove",
      "targetPath":
"$$.groups[*].display"
    },
    {
      "condition":
"$$.displayName EMPTY true",
      "type": "remove",
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",

```

```

      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User']
['validFrom']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['validTo']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:sap:2.0:User'] ['validTo']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['employeeNumber']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['costCenter']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['organization']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "targetPath": "$
['urn:ietf:params:scim:schemas:ext
ension:enterprise:2.0:User']
['division']",
      "optional": true
    },
    {
      "sourcePath": "$
['urn:ietf:params:scim:schemas:ext

```

```

    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['employeeNumber']",
    "optional": true
  },
  {
    "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['costCenter']",
    "optional": true
  },
  {
    "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['organization']",
    "optional": true
  },
  {
    "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['division']",
    "optional": true
  },
  {
    "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['department']",
    "optional": true
  },
  {
    "sourcePath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
    "targetPath": "$
['urn:ietf:params:scim:schemas:extension:enterprise:2.0:User']
['manager']['value']",
    "optional": true
  },
  {

```

```

    extension:enterprise:2.0:User']
    ['department'],
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['department'],
        "optional": true
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['value']",
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['value']",
        "optional": true
    },
    {
        "sourcePath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['displayName']",
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:enterprise:2.0:User']
    ['manager']['displayName']",
        "optional": true
    },
    {
        "sourcePath": "$.active",
        "targetPath": "$.active",
        "defaultValue": true,
        "optional": true
    },
    {
        "constant": false,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:sap:2.0:User']
    ['sendMail']",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$.emails",
        "preserveArrayWithSingleElement":
        true,
        "targetPath": "$
    ['urn:ietf:params:scim:schemas:ext
    ension:sap:2.0:User']['emails']",
        "scope": "createEntity",
        "functions": [
            {
                "function":
                "putIfAbsent",
                "key": "verified",
                "defaultValue": true
            }
        ]
    },
    {
        "function":
        "putIfAbsent",
        "key": "verified",
        "defaultValue": true
    }
]

```

```

      "sourcePath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['manager']['displayName'],
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['manager']['displayName'],
      "optional": true
    },
    {
      "sourcePath": "$
      ['urn:sap:cloud:scim:schemas:exten
      sion:custom:2.0:User'],
      "targetPath": "$
      ['urn:sap:cloud:scim:schemas:exten
      sion:custom:2.0:User'],
      "optional": true
    },
    {
      "sourcePath": "$.company",
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:enterprise:2.0:User']
      ['organization'],
      "optional": true
    }
  ],
  "group": {
    "scimEntityEndpoint":
    "Groups",
    "mappings": [
      {
        "sourcePath": "$.id",
        "targetPath": "$.id",
        "targetVariable":
        "entityIdSourceSystem"
      },
      {
        "sourceVariable":
        "entityBaseLocation",
        "targetVariable":
        "entityLocationSourceSystem",
        "targetPath":
        "$.meta.location",
        "functions": [
          {
            "type":
            "concatString",
            "suffix": "$
            {entityIdSourceSystem}"
          }
        ]
      },
      {
        "sourcePath": "$
        ['urn:sap:cloud:scim:schemas:exten
        sion:custom:2.0:Group']['name'],
        "targetPath": "$
        ['urn:sap:cloud:scim:schemas:exten
        sion:custom:2.0:Group']['name']"
      }
    ]
  }
}

```

```

      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']['emails']['*']
      ['type'],
      "type": "remove"
    },
    {
      "constant": "disabled",
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']
      ['passwordDetails']['status'],
      "scope": "createEntity"
    },
    {
      "constant": 39,
      "targetPath": "$
      ['urn:ietf:params:scim:schemas:ext
      ension:sap:2.0:User']
      ['sourceSystem'],
      "scope": "createEntity"
    },
    {
      "constant": "employee",
      "targetPath": "$.userType"
    },
    {
      "sourcePath":
      "$.timezone",
      "targetPath":
      "$.timezone",
      "optional": true
    },
    {
      "constant": "userName",
      "targetVariable":
      "entityCorrelationAttributeName"
    },
    {
      "sourcePath":
      "$.userName",
      "targetVariable":
      "entityCorrelationAttributeValue"
    },
    {
      "sourcePath":
      "$.Operations",
      "targetPath":
      "$.Operations",
      "preserveArrayWithSingleElement":
      true,
      "scope": "patchEntity"
    },
    {
      "sourcePath": "$.schemas",
      "targetPath": "$.schemas",
      "preserveArrayWithSingleElement":
      true,
      "scope": "patchEntity"
    }
  ]
}

```

```

    },
    {
      "sourcePath":
"$$.displayName",
      "targetPath":
"$$.displayName"
    },
    {
      "sourcePath":
"$$.members",
      "targetPath":
"$$.members",
      "preserveArrayWithSingleElement":
true,
      "optional": true
    },
    {
      "sourcePath": "$$.schemas",

      "preserveArrayWithSingleElement":
true,
      "targetPath": "$$.schemas"
    }
  ]
}

```

```

    },
    "group": {
      "scimEntityEndpoint":
"Groups",
      "mappings": [
        {
          "sourceVariable":
"entityIdTargetSystem",
          "targetPath": "$.id"
        },
        {
          "sourcePath":
"$$.Operations",
          "targetPath":
"$$.Operations",
          "preserveArrayWithSingleElement":
true,
          "scope": "patchEntity"
        },
        {
          "sourcePath": "$$.schemas",
          "targetPath": "$$.schemas",
          "preserveArrayWithSingleElement":
true,
          "scope": "patchEntity"
        },
        {
          "constant":
["urn:ietf:params:scim:schemas:core:2.0:Group", "urn:sap:cloud:scim:schemas:extension:custom:2.0:Group"],
          "targetPath": "$$.schemas"
        },
        {
          "sourcePath":
"$$.displayName",
          "targetPath":
"$$.displayName"
        },
        {
          "sourcePath": "$$.members",
          "targetPath": "$$.members",
          "preserveArrayWithSingleElement":
true,
          "optional": true
        },
        {
          "sourcePath":
"$$.displayName",
          "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
          "scope": "createEntity",
          "functions": [
            {
              "type":
"replaceAllString",

```

```

        "regex": "[\\s\\
\\p{Punct}]",
        "replacement": "_"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']['name']",
        "scope": "createEntity"
    },
    {
        "sourcePath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']
['description']",
        "optional": true,
        "targetPath": "$
['urn:sap:cloud:scim:schemas:extension:custom:2.0:Group']
['description']"
    }
]
}
}

```

5. Connect the external consumer to Identity Provisioning with the technical user you have created in step 1.

If the external consumer system is **SAP Identity Management**, you can export the newly created proxy system as a SCIM repository from Identity Provisioning and import it in SAP Identity Management. This will create a SCIM repository in SAP Identity Management where most of the repository constants will be automatically filled in. You need to provide the technical user credentials that you have set up in step 2 and the SCIM assignment method as described below:

- For AUTH\_USER and AUTH\_PASSWORD, enter the user ID and password of the Identity Authentication technical user for which you have set permission [Access Proxy System API](#).
- For the SCIM\_ASSIGNMENT\_METHOD constant, make sure the value is **PUT**.

### Note

For external consumer systems, other than SAP Identity Management, you should also use the **PUT** method for modifying entities.

## Next Steps

When a proxy system is connected to an external backend system (in the case of SAP Identity Management this means the exported CSV file is imported into the Identity Management Admin UI and a repository is configured), you can start managing the users and groups into this external system. Usually, the first operation

is the initial load of the existing entities into your external system. When this load has finished, changes in the external system, such as creating new users or updating existing ones, can trigger CRUD requests back to the proxy system.

To see an example with SAP Identity Management, see [Hybrid Scenario: SAP Identity Management \[page 1565\]](#) → sections **Next Steps** and **Future Identity Lifecycle**.

## Related Information

[Proxy Systems \[page 981\]](#)

[Identity Directory \[page 1567\]](#)

[Patched and Merged Attributes \[page 1590\]](#)

### 1.9.3.3 Patched and Merged Attributes

You can provision entities from multiple source systems to a single [Local Identity Directory](#) target system.

## Concept

### Note

The [Local Identity Directory](#) connector is available for both bundle and standalone tenants running on SAP Cloud Identity Services infrastructure.

If one and the same user has different personal or technical data in multiple systems, which you add as source connectors, the Identity Provisioning service will provision all the data for this user and merge it in the target identity directory.

### Caution

When reading user data from multiple source systems, we strongly recommend you perform **consecutive** provisioning jobs. Simultaneous jobs may lead to inconsistent or overwritten user data in the target system.

## Configuration

When you add the [Local Identity Directory](#) as a target system, it's configured by default, thus you don't need to enter any properties or credentials.

However, if you want to make PATCH instead of PUT requests when updating an entry, open the [Properties](#) section and add the following: `scim.support.patch.operation = true`

This extra property will provision only missing (new) attributes of a user, instead of reading the whole user data. This logic is also related to the **patchEntity** attribute in the JSON code (see section **Transformation** below). This property, along with the accordingly set target transformation, allows you to provision user attributes from multiple source systems and merge it into a single target connector.

## Transformation

### Code Syntax

```
{
  "user": {
    "mappings": [
      {
        "sourcePath": "$",
        "targetPath": "$"
      },
      {
        "targetPath": "$.id",
        "type": "remove"
      },
      {
        "sourceVariable": "entityIdTargetSystem",
        "targetPath": "$.id"
      },
      {
        "targetPath": "$.schemas",
        "type": "remove"
      },
      {
        "constant": "urn:ietf:params:scim:schemas:core:2.0:User",
        "targetPath": "$.schemas[0]"
      },
      {
        "constant":
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
        "targetPath": "$.schemas[1]"
      },
      {
        "condition": "$.emails[0].length() > 0",
        "constant": true,
        "targetPath": "$.emails[0].primary"
      },
      {
        "targetPath": "$.meta",
        "type": "remove"
      }
    ],
    // The following JSON part allows you to merge user data from multiple source
    // systems by the userName attribute (default).
    // You can also choose to merge by emails or by a SCIM Enterprise Schema
    // attribute.
    // Reminder: To enable the patchEntity operation, set property:
    scim.support.patch.operation = true
    {
      "constant": "userName",
      "targetVariable": "entityCorrelationAttributeName"
    },
    {
      "sourcePath": "$.userName",
      "targetVariable": "entityCorrelationAttributeValue"
    },
    {
      "constant": "urn:ietf:params:scim:api:messages:2.0:PatchOp",
```

```

        "targetPath": "$.schemas[0]",
        "scope": "patchEntity"
    },
    {
        "targetPath": "$.meta",
        "type": "remove",
        "scope": "patchEntity"
    },
    // After reading all users and their attributes from the first source system,
    // the Identity Provisioning starts reading users from the next one.
    // If the Identity Provisioning identifies an already existing user, it adds
    // the user's display name if this attribute was missing in the first system.
    // You can replace displayName with another attribute, or add mappings for
    // additional attributes.
    {
        "condition": "$.displayName EMPTY false",
        "constant": "add",
        "targetPath": "$.Operations[0].op",
        "scope": "patchEntity"
    },
    {
        "condition": "$.displayName EMPTY false",
        "constant": "displayName",
        "targetPath": "$.Operations[0].path",
        "scope": "patchEntity"
    },
    {
        "condition": "$.displayName EMPTY false",
        "sourcePath": "$.displayName",
        "targetPath": "$.Operations[0].value",
        "scope": "patchEntity"
    }
  ]
},
"group": {
  ...

```

If you want to merge user data by **emails** instead of **displayName**, change the transformation, modifying the following JSON lines:

#### ❖ Example

```

{
  "constant": "emails.value",
  "targetVariable": "entityCorrelationAttributeName"
},
{
  "sourcePath": "$.emails[0].value",
  "targetVariable": "entityCorrelationAttributeValue"
},

```

If you want additional user attributes from the second source system to be provisioned in the target system (for example, *nickname*), use the following pattern:

#### ❖ Example

```

{
  "condition": "$.nickName EMPTY false",
  "constant": "add",
  "targetPath": "$.Operations[0].op",
  "scope": "patchEntity"
},

```



```

{
  "condition": "$.nickName EMPTY false",
  "constant": "nickName",
  "targetPath": "$.Operations[0].path",
  "scope": "patchEntity"
},
{
  "condition": "$.nickName EMPTY false",
  "sourcePath": "$.nickName",
  "targetPath": "$.Operations[0].value",
  "scope": "patchEntity"
}

```

## Related Information

[List of Properties \[page 94\]](#)

[Identity Directory \[page 1567\]](#)

[Configuring Local Identity Directory in Target-Source Scenario \[page 1568\]](#)

## 1.10 Monitoring and Troubleshooting

### Jobs, Job Logs, and Notifications

To learn how to	See
Export or delete provisioning job logs	<a href="#">Manage Provisioning Job Logs [page 1600]</a>
Get notified by e-mail about successful or failed jobs	<a href="#">Manage Job Notifications [page 1605]</a>
Retrieve audit logs using OAuth credentials ( <i>Bundles only</i> )	<a href="#">Access Audit Logs [page 1607]</a>

## Guided Answers

The table below provides the categories of problems that you might encounter during your experience with the Identity Provisioning service. The *problem-solution* information is provided interactively in the *Guided Answers* troubleshooting tool.

## Problem Category

---

[Administration Issues](#) 

---

[Job and Transformation Issues](#) 

---

[Error Messages](#) 

---

[Additional Information](#) 

---

See also:

- [Guided Answers: Identity Provisioning Troubleshooting](#) 
- [Guided Answers: Video](#) 

## SAP Knowledge Base Articles

On the following portal, you will find a collection of all KBAs related to the Identity Provisioning service:

<https://launchpad.support.sap.com/#/products/73555000100900001028> 

To learn more about KBAs in general, see: [SAP Knowledge Base](#) 

## Get Support

If you have questions or encounter an issue while working with the Identity Provisioning service, there are various ways to address them. See: [Getting Support \[page 1620\]](#)

## 1.10.1 Monitor Provisioning Job Logs

Job logs display information about the execution of provisioning jobs. Each row in the list of job logs shows information about one execution of a job.

## Prerequisites


- You have enabled and set up a source and a target system in the Identity Provisioning UI. Make sure all mandatory properties are configured correctly. See: [Enable and Disable Systems \[page 1482\]](#)
- You have run a provisioning job. See: [Start and Stop Provisioning Jobs \[page 1524\]](#)
- Your Identity Provisioning tenant must run on SAP Cloud Identity Services infrastructure to be able to search for job logs.

## Procedure

To search and view the job logs, proceed as follows:

1. Sign in to the SAP Cloud Identity Services administration console and select ► [Identity Provisioning](#) ► [Provisioning Logs](#) ► [Job Logs](#) .  
If your tenant is running on SAP BTP, Neo environment, you access the Identity Provisioning UI and select the [Job Logs](#) tile.
2. In the [Job Execution Logs](#) screen, search for a job log and select it. You can search by system name, job type and status.

Each row in the list of job logs displays the following information:

Column	Details
<a href="#">Source System</a>	The source system the job was triggered for.
<a href="#">Job Type</a>	The job type can be <b>READ</b> or <b>RESYNC</b> .
<a href="#">Trigger Type</a>	The triggering type for the job. It can be: <ul style="list-style-type: none"><li>• <b>IMMEDIATE</b> – for manually triggered jobs after choosing <a href="#">Run Now</a> or for restarted jobs which were in previous status <b>Pending Restart</b>.</li><li>• <b>REPEATING</b> – for scheduled jobs after choosing <a href="#">Schedule</a>.</li></ul>
<a href="#">Status</a>	The status of the job. <ul style="list-style-type: none"><li>• <b>Success</b> – provisioning job has finished successfully.</li><li>• <b>Finished with Error</b> – provisioning job has finished with error. Some entities failed to be read or written, connection to the source system failed or reading from the source system encountered an issue.</li><li>• <b>Running</b> – provisioning job is still running and has not encountered any issues so far.</li><li>• <b>Running with Error</b> – provisioning job is still running, but some entities already failed.</li><li>• <b>Manually Terminated</b> – provisioning job is manually stopped by Identity Provisioning administrator.</li><li>• <b>Pending Restart</b> – provisioning job is temporary paused due to external reasons, not related to your direct interaction (for example, when Identity Provisioning is currently down). When the state gets back to normal, the job will automatically resume with a new job ID, continuing from the last processed entity. Its status will switch over to <b>Running</b> or <b>Running with Error</b>.</li></ul>
<a href="#">Start Time</a>	Date, time, and timezone in UTC format when the job is started.
<a href="#">End Time</a>	Date, time, and timezone in UTC format when the job is finished.
<a href="#">Action</a>	Allows you to stop a running provisioning job by choosing the  <a href="#">Stop Job</a> button .

3. In the *Job Execution Details* screen, view the following details about a given job:

- **<System\_Name>** – Shows the source system name, the job type, the trigger type, the job start and end time, and the job status - all described above in the *Job Execution Logs* screen. In addition, you can find information about the *Read Mode* of the job. It can be:

- **Full Read**

- **Delta Read**

The read mode considers how job-related properties, such as `ips.delta.read` and `ips.full.read.force.count`, are configured on the source system.

For example, if your system works in delta read mode and you set up `ips.full.read.force.count=10`, the *Job Execution Details* screen will display 10 consecutive jobs with *Read Mode:Delta Read* followed by one with *Read Mode:Full Read*.

### Note

**Resync** and **Simulate** jobs (the latter is available for Identity Provisioning tenants on SAP Cloud Identity Services infrastructure) will always have Read Mode: Full Read. Migration jobs does not display read mode.

- **Error Message** – Shows the error message.
- **Statistics** – Shows the job statistics, that is, details about how entities are handled.

Column	Source System	Target System
<i>Entity</i>	Type of the entity (user, group)	Type of the entity (user, group)
<i>System</i>	Name of the source system	Name of the target system
<i>Action</i>	Action executed on the system: <i>Read</i>	Action executed on the system: <i>Write</i>
<i>Read</i>	Number of read entities	Not applicable
<i>Created</i>	Not applicable	Number of created entities
<i>Updated</i>	Not applicable	Number of updated entities
<i>Deleted</i>	Not applicable	Number of deleted entities

Column	Source System	Target System
<i>Skipped</i>	<p>Number of skipped entities from the source system</p> <p>Entities can be skipped if they don't fulfill a condition in the read transformation.</p> <div> <p>❖ Example</p> <p>Setting this group condition in the read transformation results in skipping all read groups that do not match the specified display name.</p> <pre> ❏ Code Syntax "condition": "\$\$.displayName == 'Employee' ", </pre> </div>	<p>Number of skipped entities in the target system</p> <p>Entities can be skipped if they don't fulfill a condition in the write transformation.</p> <div> <p>❖ Example</p> <p>Setting this group condition in the write transformation results in skipping all groups that do not match the specified display name.</p> <pre> ❏ Code Syntax "condition": "\$\$.displayName == 'Employee' ", </pre> </div>
<i>Failed</i>	<p>Number of entities failed during the read operation from the source system</p>	<p>Number of entities failed to be created, updated or deleted in the target system</p>

- **Failed Entities** – Shows additional information about the failed entities. A maximum number of ten entities can be displayed per job log. In case you need to view the full job log, download it from the



[Download All Error Logs for This Job](#) button. For more information see: [Manage Provisioning Job Logs \[page 1600\]](#)

## Related Information

[Start and Stop Provisioning Jobs \[page 1524\]](#)

[Manage Job Notifications \[page 1605\]](#)

## 1.10.2 Monitor Real-Time Logs

Real-time provisioning logs display information about the execution of a real-time sync of a user or a group entity.

### Prerequisites

- Your Identity Provisioning tenant is running on SAP Cloud Identity infrastructure.
- You have configured the real-time provisioning scenario. For more information, see [Real-Time Provisioning in SAP Cloud Identity Infrastructure \[page 1557\]](#)
- Your source and target systems are enabled. For more information, see [Enable and Disable Systems \[page 1482\]](#)

### Procedure

You can search, view, refresh and configure a retention period of real-time provisioning logs. Proceed as follows:

1. Sign in to the SAP Cloud Identity Services administration console and select ► [Identity Provisioning](#) ► [Provisioning Logs](#) ► [Real-Time Logs](#) .
2. In the [Real-Time Provisioning Execution Logs](#) screen, search for a log and select it. You can search by source system name, entity ID, entity type and status.  
Each row in the list of logs displays the following information about one execution of a real-time sync.

Column	Details
<a href="#">Source System</a>	<p>The name of the source system the real-time provisioning is triggered from.</p> <p>For example: <a href="#">IAS_Source</a></p>
<a href="#">Entity ID</a>	<p>The entity ID of the user or the group in the target system.</p> <p>For example: <a href="#">baba3ba7-35f0-4567-b507-ce5555c11bbb</a></p> <p>If the entity ID is missing, this means that the real-time provisioning has failed. It happens when something goes wrong with the source system.</p> <p>For example, the source system is disabled or establishing the connection to the source system has failed due to incorrect URL. In this case, you get the log without the entity ID.</p>
<a href="#">Entity Type</a>	<p>The entity type: user or group.</p>

Column	Details
<a href="#">Start Time</a>	Date, time, and time zone in UTC format when the real-time sync is triggered.  For example: <i>28/Jun/2023 09:19:00 +03</i>
<a href="#">Status</a>	The status of the real-time provisioning.  For example: <ul style="list-style-type: none"> <li>• <i>Finished Successfully</i> - Real-time provisioning finished successfully. A user or a group is successfully created, updated or deleted real time.</li> <li>• <i>Finished with Error</i> - Real-time provisioning finished with error. A user or a group failed to be created, updated or deleted real time.</li> </ul>

3. In the [Real-Time Log Details](#) screen, view the following details under the [Source System](#) name: log type ([Real Time](#)), entity ID (if it is available), start time, status, error message and [Result on Target System](#) table.

Column	Details
<a href="#">Target System</a>	The name of the target system the real-time provisioning is triggered to.  For example: <i>Concur_Target</i>
<a href="#">Operation</a>	The type of the operation. <ul style="list-style-type: none"> <li>• <i>CREATE</i></li> <li>• <i>UPDATE</i></li> <li>• <i>PATCH</i></li> <li>• <i>DELETE</i></li> </ul>
<a href="#">State</a>	The state of the entity when real-time provisioning is triggered. <ul style="list-style-type: none"> <li>• <i>Entity Created</i></li> <li>• <i>Entity Updated</i></li> <li>• <i>Entity Skipped</i></li> <li>• <i>Entity Already Provisioned</i></li> <li>• <i>Entity Failed</i></li> <li>• <i>Entity Read</i></li> <li>• <i>Entity Deleted</i></li> <li>• <i>Entity Not Provisioned</i></li> </ul>
<a href="#">Additional Information</a>	Additional information about the failed entity.

4. (Optional) Choose [Refresh](#) to get the latest updates of the real-time provisioning logs.
5. Choose [Configure](#) to define the retention period of real-time provisioning logs.  
Set a period (7, 14 or 30 days). Logs which are older than this period will be automatically deleted. By default, job logs are kept for 7 days.

## ! Restriction

Note the following restrictions about the real-time provisioning logs:

- You cannot delete them manually.
- You cannot download them.
- You cannot subscribe to a source system to receive notifications about their status.


## 1.10.3 Manage Provisioning Job Logs

After you view and analyze the provisioning job logs, you can download or delete them.


### Prerequisites

- You have enabled and set up a source and a target system in the Identity Provisioning UI. Make sure all mandatory properties are configured correctly. See: [Enable and Disable Systems \[page 1482\]](#)
- You have run a provisioning job. See: [Start and Stop Provisioning Jobs \[page 1524\]](#)

### Download Execution Logs for All Jobs

1. Access the Identity Provisioning UI and select **Provisioning Logs** > **Job Logs**.  
If your tenant is running on SAP BTP, Neo environment, you access the Identity Provisioning UI and select the **Job Logs** tile.
2. From the upper right corner, choose  **Download Execution Logs for All Jobs** button.
3. Choose **Download**. If the number of logs is too large, the execution logs will be downloaded in parts. Each part (a ZIP archive) contains 3000 logs, by default.
4. Save all ZIP files in your local file system.

### Download All Error Logs for a Single Job

1. Access the Identity Provisioning UI and select **Provisioning Logs** > **Job Logs**.  
If your tenant is running on SAP BTP, Neo environment, you access the Identity Provisioning UI and select the **Job Logs** tile.
2. Select a job that has finished with error.
3. Above the **Failed Entities** table, choose the  button and select **Download All Error Logs for This Job**.



4. Choose [Download](#). The job log is downloaded as a ZIP archive. Its name pattern is: **ips\_jobErrorLogs\_<job ID>\_<date>\_<time>**  
**Hint:** The job ID is at the end of the URL of the selected job in the Identity Provisioning admin console.
5. Save the ZIP file in your local file system, and then open it to view the log records of all failed entities from this job.

#### **i** Note

If your provisioning job is in **Pending Restart** status (that is, temporary paused due to external reasons), and is already running with error, only the logs up until the pausing moment will be printed in the current ZIP file. The rest of the logs will be printed in the next ZIP file - when the job is resumed and finished.

If you manually stop a failing job, only the error logs up until that moment will be printed in the ZIP file.

## Download All Skipped Entity Logs for a Single Job

#### **i** Note


This functionality is available only for Identity Provisioning bundle and standalone tenants running on SAP Cloud Identity infrastructure.

### Prerequisites

- You have set the `ips.trace.skipped.entity` property to **true** in the source system.
- You have set the `ips.trace.skipped.entity.content` property to **true** in the source system.

For more information, see [List of Properties \[page 94\]](#)

Download the skipped entities for a single job to identify the entities themselves, the systems they are skipped from, the reason behind this, as well as to view the content of the entities.

1. Access the Identity Provisioning UI and select ► [Provisioning Logs](#) ► [Job Logs](#) ►.
2. Select a job that has at least one skipped entity.
3. On the right-hand side of the [Statistics](#) section, choose the  button and select [Download All Skipped Entity Logs for This Job](#).  
The log is downloaded as a zip archive. The name of the file follows the pattern:  
`ips_jobSkippedEntitiesLogs_<job ID>_<date>_<time>`.
4. Save the zip file in your local file system, and then open it to view the log records of all skipped entities for this job.

The log displays the following information:

Entity Details	Value
<i>User</i>	The ID of the skipped entity
<i>Group</i>	For example: <b>12345678-1a2b-1bc2-3cd4-1234567890ef</b>
<i>System</i>	The system the entity is skipped from. This could be either a source system or a target system.  For example: <b>IAS.target</b>
<i>Skip Reason</i>	<p>The reason the entity is skipped</p> <p>Normally, a user or a group is skipped because it does not fulfill a condition in the source read transformation or the target write transformation. If this is the case, you will see the exact condition the entity does not fulfill.</p> <p>For example:</p> <div><p>≡ Code Syntax</p><pre>The condition: (\$.emails EMPTY false) &amp;&amp; (\$.userName EMPTY false), is not fulfilled.</pre></div> <p>A user or a group can also be skipped if you use the <code>skipOperations</code> expression in the transformations to avoid creating, deleting, or updating entities in target systems. If this is the case, you will see the following message: Operation [<b>&lt;create&gt;&lt;update&gt;&lt;delete&gt;</b>] is skipped in target transformation</p>

## Entity Details

## Value

### Content

The content (attributes of the skipped entity)

For example:

#### Code Syntax

```
{ "active": true, "displayName": "John Smith", "emails": [ { "value": "john.smith@example.com" } ], "name": { "familyName": "Smith", "givenName": "John" }, "schemas": [ "urn:ietf:params:scim:schemas:core:2.0:User", "urn:ietf:params:scim:schemas:extension:sap:2.0:User" ], "urn:ietf:params:scim:schemas:extension:sap:2.0:User": { "userUuid": "0036024b-0ede-4fc3-9ed7-c55632de8246" }, "urn:sap:cloud:scim:schemas:extension:custom:2.0:User": { "userId": "12345678-1a2b-1bc2-3cd4-1234567890ef", "userName": "", "userType": "public" }
```



The log is organized in sections which start with the ID of the skipped entity. If a user or a group is skipped in more than one systems, the log will display the ID of the skipped entity as many times as there are systems where the entity is skipped from, the skip reason and the entity's content.

## Download Job Logs via API

### Note

This functionality is available only for Identity Provisioning bundle and standalone tenants running on SAP Cloud Identity infrastructure.

- You need a technical user (a user of type System in SAP Cloud Identity Services admin console) with [Access Identity Provisioning Tenant Admin API](#) permission assigned. For more information, see [Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#)

Use the Identity Provisioning tenant admin API to download job logs via API calls for a single provisioning job. The API is available on the SAP Business Accelerator Hub: [SAP Cloud Identity Services](#)  [Identity Provisioning Service](#) > [API Reference](#) > [JobLogs](#) . The URL for accessing the Tenant Admin API

follows the pattern: `https://<ias-tenant-host>/ips/service/publicapi/v1/jobLogs/<JobId>?action=export&logType=<logType>`, where:

Parameter	Value
<code>&lt;JobId&gt;</code>	The job identifier
Action	The action to be executed Only possible value: <b>export</b>
<code>&lt;logType&gt;</code>	The type of the job log to be downloaded Possible values: <ul style="list-style-type: none"><li>• <b>commonLog</b> - Download logs containing the following details for this specific job execution (start time, end time, statistics, type, mode). These details are currently included in the common logs for all job executions.</li><li>• <b>failedEntitiesLog</b> - Download logs containing details about the failed entities for this specific job execution.</li><li>• <b>skippedEntitiesLog</b> - Download logs containing details about the skipped entities for this specific job execution.</li></ul>

For example: `https://<ias-tenant-host>/ips/service/publicapi/v1/jobLogs/cbc1eb01686664856111?action=export&logType=allErrorsLog`

Save the zip file in your local file system, and then open it to view the log records for the log type you have defined.

## Delete Job Logs

If you don't need your job logs anymore, you can delete them. You can do this manually or automatically (by setting a retention period).


1. Access the Identity Provisioning UI and select **Provisioning Logs** > **Job Logs**.  
If your tenant is running on SAP BTP, Neo environment, you access the Identity Provisioning UI and select the **Job Logs** tile.

2. Choose the  **Delete Logs** icon.

### ⚠ Caution

This deletes the logs for all finished jobs. If a job is still running though, it will stay along with its logs.

3. You can set a time period for keeping the job logs available for monitoring.

1. From the upper right corner, choose  **Configure Job Logs Settings**.

2. Set a period (**7**, **14** or **30** days). Logs which are older than this period will be automatically deleted. By default, job logs are kept for 7 days.
3. If you want to keep the logs longer, you can download and save them in your local file system.

## Related Information

[Monitor Provisioning Job Logs \[page 1594\]](#)

[Security: Job Logs \[page 1548\]](#)

## 1.10.4 Manage Job Notifications

You can subscribe to a source system to receive notification e-mails. They provide information about the job execution and links to download job logs, in case of failed or skipped entities.

### Prerequisites

Before subscribing yourself to a source system, make sure that the trust between your Identity Authentication and SAP BTP is properly set, otherwise the Identity Provisioning might not propagate your email address. To do that:

1. Open SAP BTP cockpit and navigate to your Neo subaccount.
2. Navigate to ► [Security](#) ► [Trust](#) ► [Application Identity Provider](#) ►.
3. Check your current identity provider.
  - If it's the default one – **SAP ID Service** (<https://accounts.sap.com>), you don't need to do anything.
  - If it's a custom one, choose the link to open it for edit. Go to tab [Attributes](#), and then follow steps **11–12** on page: [Setting Up Trust Between Identity Authentication and SAP BTP](#)

### Context

When you subscribe to a source system, you can receive notification e-mails in the following cases:

- You start or schedule a provisioning job and it fails.  
You'll receive an e-mail with subject *Provisioning Running with Error. Source System: <name>*. You receive one e-mail per job, after the first failed entity. If more entities fail during this job, no additional e-mails will be sent. This behavior is related to property `ips.job.notification.skip.intermediate.notifications`.
- The failed job has finished.  
You'll receive an e-mail with subject *Provisioning Finished with Error. Source System: <name>*. By default, if the same job runs again and keeps failing, no

further notifications will be sent to your e-mail. However, you can control the notifications via properties `ips.job.notification.ignored.consecutive.failures` and `ips.job.notification.repeat.on.failure`. For more information, see: [List of Properties \[page 94\]](#)

- The job is back to normal (the problem with the failed entities has been resolved).  
After a new run, the job has successfully finished. You'll receive only one e-mail with subject *Provisioning Success. Source System: <name>*.

The notification e-mail tells you which are the source and the target systems, what is the job type, its start time and status. It contains the `Navigate to details` link that opens the *Job Execution Details* screen in the Identity Provisioning UI.

In case your job finished with failed or skipped entities, links to `Download error logs` and `Download skipped entities logs` are provided, respectively. The content of the downloaded log files depends on your configuration of the `ips.trace.skipped.entity`, `ips.trace.skipped.entity.content` and `ips.trace.failed.entity.content` properties. For more information, see: [List of Properties \[page 94\]](#)

#### **i** Note

Sending navigation links to the download page of failed and skipped entities is supported for tenants running on SAP Cloud Identity Services infrastructure.

#### **i** Note


If you subscribe to a source system, and then run a successful provisioning job, no notification e-mails will be sent.

## Procedure


1. Access the Identity Provisioning UI and choose the *Source Systems* tile.
2. Select the system you need to watch and choose *Jobs*.




#### **i** Note

Notifications are supported only for read and resync jobs.

3. From the bottom right corner, choose *Subscribe*.
  - To subscribe yourself, choose *Subscribe me*.
  - To subscribe another user or a group (distribution list), choose *Subscribe others*. Fill in the required fields and choose  *Add*.

#### **i** Note

From the *Recipients* list, you can remove existing subscribers. To do that, go to the *Action* column and choose the  icon.

4. You can now run or schedule a provisioning job.
5. If you no longer need to be subscribed to a source system, choose  *Subscribe*  *Unsubscribe me* .

## Related Information

[Monitor Provisioning Job Logs \[page 1594\]](#)


### 1.10.5 Access Audit Logs

You can access audit logs to track changes for events and activities in your Identity Provisioning tenant.

#### Tenants on SAP BTP Neo Environment

This procedure is applicable only for bundle tenants on SAP BTP Neo environment. To check the list of cloud products that include Identity Provisioning and Identity Authentication (free of charge), see: [Obtain a Bundle Tenant \[page 407\]](#).

To view the audit logs, you have to first generate Client ID and Client Secret in the Identity Provisioning user interface. Use these credentials to obtain an access token, and then call the audit log retrieval API. Follow the procedure below.

1. From the Identity Provisioning UI home page, go to the [Security](#) section and choose the [OAuth](#) tile.
2. Choose  [Generate Credentials](#).
3. Enter a description for your OAuth client or leave the field empty.
4. Choose [Save](#). A pop-up with generated credentials appears.

#### → Remember

Copy and save the [Client Secret](#) as you won't be able to retrieve it later.

5. The [Client ID](#) appears in the OAuth table.

#### i Note

You are only allowed to use a single set of OAuth client credentials. If you want to use another credentials, delete the old ones and generate a new set.

6. Use the generated credentials to obtain an access token. To learn how, see [Using Platform APIs → 2. Get an OAuth Access Token](#)
7. You can now retrieve audit logs. To learn how, see: [Audit Log Retrieval API Usage for the Neo Environment](#).

#### i Note

The OAuth clients generated by the Identity Provisioning service have only the [Read Audit Logs](#) scope assigned. That means, you can only read (retrieve) audit logs but you cannot modify their retention period.

The default retention period is [201](#) days.

For more information, see: [Audit Logs: Retention and Retrieval APIs](#)

## SAP Cloud Identity Services Tenants

This section is applicable only for SAP Cloud Identity Services tenants running on the AWS and Azure infrastructure.

To enable the audit logs in SAP Cloud Identity Services administration console, proceed as follows:

1. Follow the procedures described in [Access Audit Logs \(AWS, Azure Infrastructure\)](#).
2. Open the link provided under the ► [Monitoring & Reporting](#) ► [Audit and Change Logs](#) ► [Auditlog Viewer](#) ►.
3. Enter a date range and a filter for the audit logs that you want to view.

### Note

No records are displayed for events that occurred prior to enabling the audit logs configuration.

The default retention period is [90](#) days.

The audit logs provide information about the event category and timestamp, the event and object type, who performed the action and others. For example:

- **Category:** audit.security-events (logged as security event message), audit.configuration (logged as configuration modification message), audit.data-access (logged as data access message).
- **Event Type:** JOB\_TRIGGERED, SYSTEM\_UPDATED, SYSTEM\_CREATED, SYSTEM\_DELETED
- **Object Type:** Job, System
- **ObjectAttribute.performed-by-user:** P123456

## 1.11 Standalone Tenants

A standalone tenant allows you to use Identity Provisioning as a separate (standalone) product.

### ⚠ Caution

**Effective October 20, 2020, Identity Provisioning is offered bundled with SAP cloud solutions.** You can obtain and use it, along with Identity Authentication, as part of a bundled SAP cloud solution that you need to purchase. The service is no longer sold as a standalone product. Existing customers of standalone Identity Provisioning can use it as-is until the end of their contracts.

To check the list of SAP cloud solutions that bundle Identity Provisioning, see [Bundle Tenants and Connectors \[page 422\]](#)

The standalone tenant can be used for provisioning user data to and from all supported systems by Identity Provisioning service. The table below lists the provisioning systems (connectors) which are available as source, target and proxy systems in standalone tenants.

Connector Type	Source System	Target System	Proxy System
<a href="#">Identity Authentication</a>	✓	✓	✓



Connector Type	Source System	Target System	Proxy System
Local Identity Directory (Not supported on Neo tenants)	✓	✓	✓
SAP Analytics Cloud	✓	✓	✓
SAP Application Server ABAP	✓	✓	✓
SAP Ariba Applications	✓	✓	✓
SAP BTP ABAP environment	✓	✓	✓
SAP BTP Account Members (Neo)	✓	✓	✓
SAP BTP Java/HTML5 apps (Neo)	✓	✓	✓
SAP BTP XS Advanced UAA (Cloud Foundry)	✓	✓	✓
SAP Build Work Zone, advanced edition	✓	✓	✓
SAP Central Business Configuration	✓	✓	✓
SAP Commissions	✓	✓	✓
SAP Concur	✓	✓	✓
SAP CPQ	✓	✓	✓
SAP Enterprise Portal	✓		
SAP Fieldglass	✓	✓	✓
SAP Integrated Business Planning for Supply Chain	✓	✓	✓
SAP Jam Collaboration	✓	✓	✓
SAP Marketing Cloud	✓	✓	✓
SAP Master Data Integration	✓		✓

Connector Type	Source System	Target System	Proxy System
<i>SAP S/4HANA Cloud</i>	✓	✓	✓
<i>SAP S/4HANA On-Premise</i>	✓	✓	✓
<i>SAP Sales Cloud and SAP Service Cloud</i>	✓	✓	✓
<i>SAP SuccessFactors</i>	✓	✓	✓
<i>SAP SuccessFactors Learning</i>	✓		
<i>Sales Cloud – Analytics &amp; AI</i>	✓	✓	✓
<i>Cloud Foundry UAA Server</i>	✓	✓	✓
<i>Google G Suite</i>	✓	✓	✓
<i>LDAP Server</i>	✓	✓	✓
<i>Microsoft Active Directory</i>	✓	✓	✓
<i>Microsoft Azure Active Directory</i>	✓	✓	✓
<i>SCIM System</i>	✓	✓	✓
<i>SSH Server (Beta)</i>	✓	✓	
<i>SAP HANA Database (Beta)</i>		✓	
<i>SAP Document Center</i>		✓	

## Initial Setup

The initial set up of your Identity Provisioning standalone tenant involves the following steps:

1. Order a monthly subscription to Identity Provisioning. For more information, see [Use a Standalone Tenant \[page 1612\]](#).
2. Log on to the Identity Provisioning user interface (UI). For more information, see [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)
3. Set up the source, target, and proxy systems for your provisioning scenario.  
In standalone tenants, source and target systems are not configured. You need to add them in the Identity Provisioning UI. See [Add a System \[page 1477\]](#)

When adding a system, Identity Provisioning works as follows:

- Identity Provisioning in **default mode**: Provision user data from source to target systems. You add source and target systems only. This could be one source connected to one or multiple target systems, or one target connected to one or multiple source systems. For more information, see: [Source Systems \[page 452\]](#) and [Target Systems \[page 702\]](#).
  - Identity Provisioning in **proxy mode**: Provision user data to and from a central identity management solution and a system with proxy configuration. You add a proxy system and you have an external identity management system (such as, SAP Identity Management) in place. For more information, see: [Proxy Systems \[page 981\]](#).
4. Configure the connection details for your systems. You have the following options:
- Add properties in the Identity Provisioning UI and provide the required connection information. For more information, refer to the respective provisioning systems (connectors) listed under [Supported Systems \[page 452\]](#) section where mandatory properties are specified.
  - Create a destination in your subaccount in SAP BTP cockpit and select it for the given provisioning system in the Identity Provisioning UI.
- Recommendation

Creating a destination is mandatory for configuring SAP Application Server ABAP provisioning systems and on-premise systems using the Cloud Connector for which a [Location ID](#) is configured. You can also use it if you need to reuse one and the same configuration for multiple provisioning systems. In all other cases, we recommend that you use the [Properties](#) tab.
5. Define what data you want to provision. You have the following options:
- Adapt the default transformation logic or use it as-is. For more information, see: [Transformations \[page 323\]](#).
  - Configure filtering properties for users and groups. For more information, see: [Properties \[page 90\]](#).
6. Run a provisioning job manually or set a time interval for scheduled jobs. For more information, see: [Start and Stop Provisioning Jobs \[page 1524\]](#).

## Viewing Your Tenants

As an SAP customer, you can view all your Identity Authentication and Identity Provisioning tenants by accessing the [SAP Cloud Identity Services - Tenants](#) application at the following URL: <https://iamtenants.accounts.cloud.sap/>. It displays the type of the tenant (test or productive), the date it was created, the region where it is available and the tenant administrators. For more information about the data you can view and how to log on, see:

- [Viewing Assigned Tenants and Administrators](#)
- [New! Check on one single page all of your Identity Authentication and Identity Provisioning tenants and administrators](#)

## Related Information

[Tenant Model \[page 8\]](#)

[Tenant Infrastructure \[page 10\]](#)

[Use a Standalone Tenant \[page 1612\]](#)

[Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

## 1.11.1 Use a Standalone Tenant

You can use the Identity Provisioning service as a standalone product.

### Context

Standalone tenants can run on SAP Cloud Identity Services infrastructure and SAP BTP, Neo environment.

- Identity Provisioning service purchased between September 1, 2020, and October 20, 2020, runs on the infrastructure of SAP Cloud Identity Services.
- Identity Provisioning service purchased before September 1, 2020 runs on SAP BTP, Neo environment.

## SAP Cloud Identity Infrastructure

You can use an account for Identity Provisioning in two ways:

- You already have a global account for SAP BTP. That means, you also have a tenant for Identity Authentication. In this case, you can use the same tenant for Identity Provisioning as well.
- You don't have any accounts for SAP BTP. In this case, after purchasing the Identity Provisioning, you will receive a global SAP BTP account, from which you can access both Neo and Cloud Foundry environments via the SAP BTP cockpit. You will also receive a tenant ID, which you can use for both Identity Authentication and Identity Provisioning.

### Procedure

1. Order a monthly subscription to Identity Provisioning.
2. After you have purchased a subscription for Identity Provisioning, you will receive an e-mail. It contains your tenant ID, which gives you access to both Identity Authentication and Identity Provisioning, on the following respective URLs:
  - `https://<tenant_id>.accounts.ondemand.com/admin`
  - `https://<tenant_id>.accounts.ondemand.com/ips`
3. Confirm the registration of your first user. This user will receive administration rights for the tenant.

## Next Steps

You can now open the Identity Provisioning user interface to start working with the service. See: [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

## SAP BTP, Neo Environment

If you don't have an SAP BTP global account, you will obtain one when you get the Identity Provisioning. If you already have a global account, you can consume your purchased Identity Provisioning with your existing subaccounts.

### Procedure

1. You have ordered a monthly subscription to Identity Provisioning.
2. After you have purchased a subscription for Identity Provisioning, you have received an e-mail. It contains a link to your SAP BTP global account in SAP BTP cockpit, for which Identity Provisioning is activated.
3. Confirm the registration of your first user. This user will receive administration rights for this global account, and will be the initial administrator for the Identity Provisioning administration console.

#### → Tip

As a next step, we recommend that you create two subaccounts and enable the Identity Provisioning for both of them.

- Use the first subaccount for **testing** purposes only, to see how the service works. For example, you can configure internal systems and run jobs to provision fake entities. If a job fails, this will not affect your real entities and productive systems.
- When your systems are correctly configured and jobs run successfully, you can then open the Identity Provisioning UI at your second subaccount to execute **productive** scenarios. To avoid double work, export the existing configured systems from your test subaccount and import them in your productive one.

For more information, see [Create a Subaccount](#) and [Export and Import Systems \[page 1482\]](#).

## Next Steps

You can now open the Identity Provisioning user interface to start working with the service. See: [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#)

## 1.11.2 Access Identity Provisioning UI of Standalone Tenants

Access the Identity Provisioning user interface as a standalone product.

### Prerequisites

You have ordered a monthly subscription for the Identity Provisioning service. See: [Use a Standalone Tenant \[page 1612\]](#)

## SAP Cloud Identity Infrastructure

### Context

Ensure your tenant is running on SAP Cloud Identity Services infrastructure. For more information, see [Tenant Model \[page 8\]](#)

### Procedure

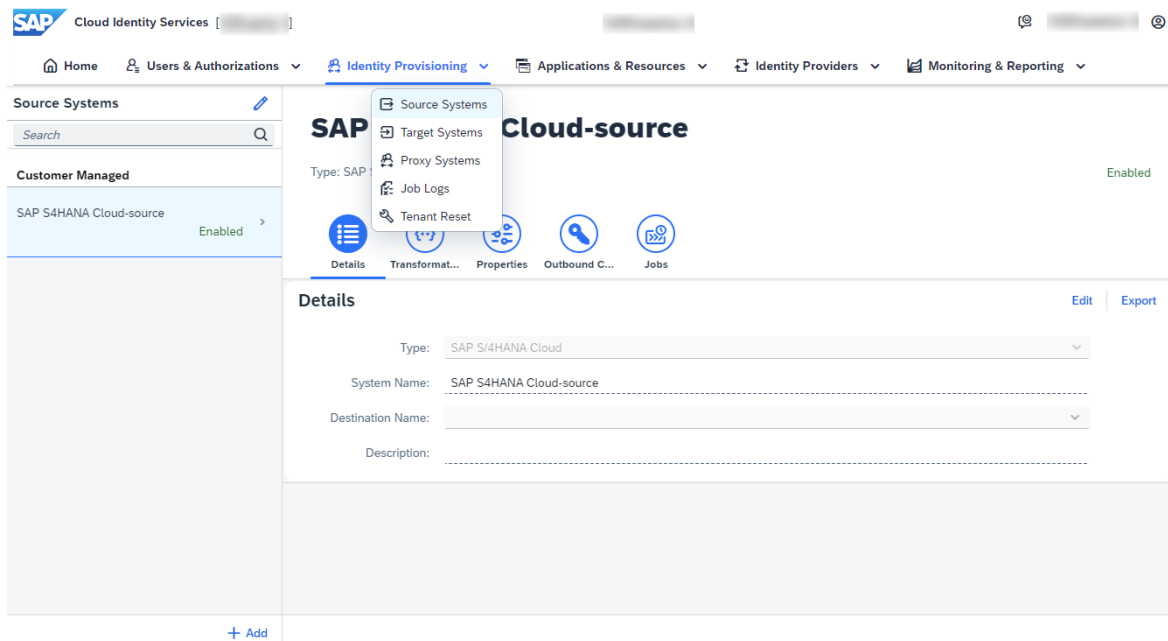
1. You have received the URL for accessing the SAP Cloud Identity Services administration console. The S-user that has been provided for creating the account is the initial administrator of this Identity Provisioning tenant.

The URL follows the pattern: `https://<ias-host>/admin`, where the Identity Provisioning tenant URL uses the host of the corresponding Identity Authentication tenant of the customer.

If your tenant URL is `https://<ias-host>/ips`, you are redirected to `https://<ias-host>/admin`. This opens the common administration console of SAP Cloud Identity Services, where the provisioning functionality is embedded under *Identity Provisioning* section.

2. Navigate through Identity Provisioning.

Your starting point is the list of *Source Systems* in the administration console of SAP Cloud Identity Services. The entire provisioning functionality can be accessed through the navigation area under *Identity Provisioning* → *Source Systems*, *Target Systems*, *Proxy Systems*, *Job Logs* and *Tenant Reset*.



- (Optional) Add additional users as tenant administrators and create a technical user with the necessary authorizations for configuring real-time provisioning and proxy systems.

For more information, see [Manage Authorizations in SAP Cloud Identity Infrastructure \[page 1487\]](#)

## Next Steps

- To configure source and target systems, and run provisioning jobs, see [Supported Systems \[page 452\]](#) and [Start and Stop Provisioning Jobs \[page 1524\]](#).
- To configure on-premise systems, see [Connect to On-Premise Systems in SAP Cloud Identity Infrastructure \[page 1512\]](#)
- To configure real-time provisioning, see [Real-Time Provisioning in SAP Cloud Identity Infrastructure \[page 1557\]](#)
- To get support and open an incident if you encounter issues while configuring the Identity Provisioning service, see [Getting Support \[page 1620\]](#). You can also ask a question in the SAP Community.

## SAP BTP, Neo Environment

### Context

Ensure your tenant is running on SAP BTP, Neo environment. For more information, see [Tenant Model \[page 8\]](#)

## Procedure

1. Open the SAP BTP cockpit. The [Overview](#) section is displayed by default.

For more information, see [SAP BTP Cockpit](#) and [Regions and Hosts \(Neo\)](#).

2. Select your region and then your global account.

### → Tip

As a next step, we recommend that you create two subaccounts and enable the Identity Provisioning for both of them.

- Use the first subaccount for **testing** purposes only, to see how the service works. For example, you can configure internal systems and run jobs to provision fake entities. If a job fails, this will not affect your real entities and productive systems.
- When your systems are correctly configured and jobs run successfully, you can then open the Identity Provisioning UI at your second subaccount to execute **productive** scenarios. To avoid double work, export the existing configured systems from your test subaccount and import them in your productive one.

For more information, see [Create a Subaccount](#) and [Export and Import Systems \[page 1482\]](#).

3. Create and save your subaccounts. They appear in the [Subaccounts](#) list.

### ! Restriction

By default, you are entitled to activate the Identity Provisioning on **two** subaccounts per global account. If your business needs require using the service on more subaccounts, create an incident to component **BC-IAM-IPS** and request a rise of your quota, specifying the number of additional subaccounts.

4. Open your subaccount and navigate to [Services](#).
5. From the [Extension Suite – Development Efficiency](#) section, choose the [Identity Provisioning](#) tile.
6. The default status of the service is **Not enabled**. Choose [Enable](#) to make it available for work.
7. (Optional) You can assign administrator permissions to additional users from your company.
  - a. Click [Configure Service](#).
  - b. On the left-side menu, choose [Roles](#). The first table shows that the **IPS\_ADMIN** role is assigned to your user by default.
  - c. Go to the second table and choose the [Assign](#) tab.
  - d. Enter the user ID of the additional corporate user. For example, [P123456789](#) (case insensitive). You can add as many additional users as you need.
  - e. Choose [Assign](#). The relevant user ID is added to the second table, and the **IPS\_ADMIN** role is assigned to this user.

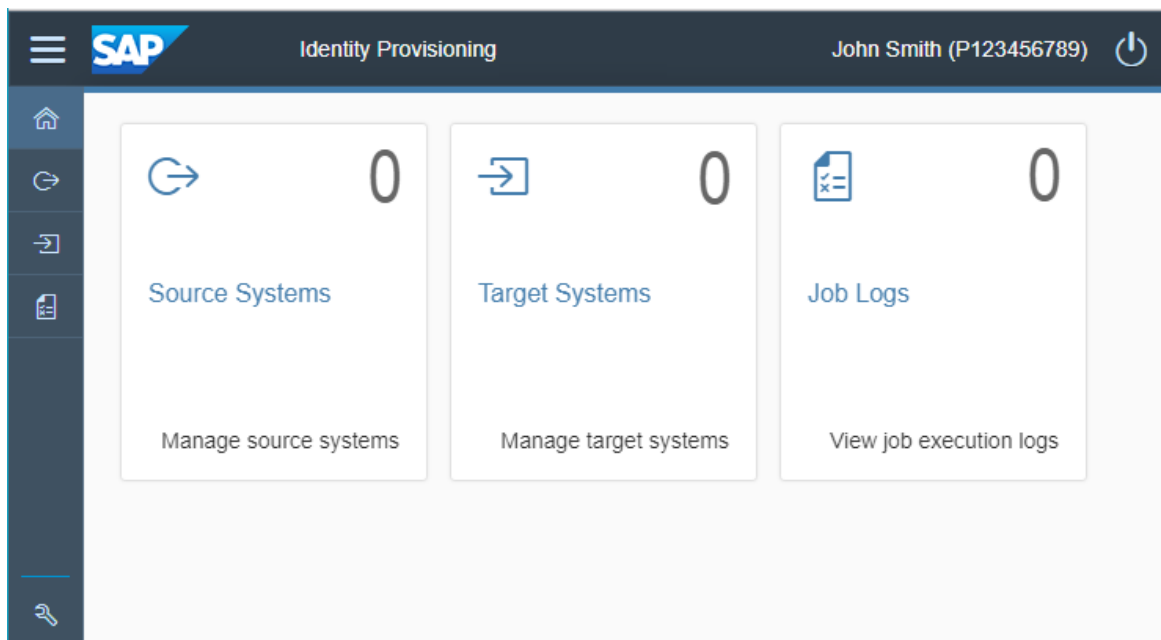
For more information, see [Manage Authorizations in Neo Environment → Standalone Tenants \[page 1490\]](#)

8. From the breadcrumbs path, choose [Identity Provisioning](#)→[Go to Service](#).
9. (Optional) If you later go to the navigation area and open [Applications](#) > [Subscriptions](#), your subaccount should be subscribed to the following provider applications: [ips](#) (Java and HTML5), [idds](#) (Java), and, if requested, [ipsproxy](#) (Java).

You can access the Identity Provisioning UI from the HTML5 application on the following URL: `https://ips-<consumer_account>.dispatcher.<region_host>`



10. The Identity Provisioning UI opens as an independent HTML5 application. The Home section displays the following tiles: *Source Systems*, *Target Systems*, *Proxy Systems* and *Job Logs*.



#### i Note

Secure communication is provided between this HTML5 application and the SAP BTP cockpit, realized by principal propagation. This process is automatically enabled by a back-end script.

## Next Steps

- To configure source and target systems, and run provisioning jobs, see [Supported Systems \[page 452\]](#) and [Start and Stop Provisioning Jobs \[page 1524\]](#).
- To configure on-premise systems, see [Connecting to On-Premise Systems in Neo Environment \[page 1516\]](#)
- To configure real-time provisioning, see [Real-Time Provisioning in Neo Environment \[page 1560\]](#)
- To get support and open an incident if you encounter issues while configuring the Identity Provisioning service, see [Getting Support \[page 1620\]](#). You can also ask a question in the SAP Community.

## 1.12 Service Offboarding

This section explains how can you terminate or deactivate your Identity Provisioning service.

You can deactivate or terminate your productive Identity Provisioning service depending on your purchase type. To learn more, see:

- [Reset the Identity Provisioning \(Bundles\) \[page 1618\]](#)
- [Reset/Remove the Identity Provisioning \(Standalone\) \[page 1618\]](#)

## 1.12.1 Reset the Identity Provisioning (Bundles)

### Context

#### ! Restriction

This operation is applicable only for bundle accounts. To check the list of cloud products that include Identity Provisioning and Identity Authentication (free of charge), see: [Obtain a Bundle Tenant \[page 407\]](#)

If you no longer need to use the Identity Provisioning service, you can deactivate it by resetting it to its default state. That means, all provisioning systems and jobs will be deleted.

To learn how, see: [Reset Identity Provisioning Tenant \[page 1542\]](#)

## 1.12.2 Reset/Remove the Identity Provisioning (Standalone)

If you want to terminate your work with the Identity Provisioning service, you can do it in two ways. Choose the one that suits your use case.

### Reset your Identity Provisioning

If you no longer need the Identity Provisioning service but want to keep it for future use, you can deactivate it by resetting it to its default state. That means, all provisioning systems and jobs will be deleted.

To learn how, see: [Reset Identity Provisioning Tenant \[page 1542\]](#)

### Remove your Identity Provisioning subscriptions

1. Sign in to the SAP BTP cockpit: <https://account.hana.ondemand.com/>
2. Go to your Identity Provisioning subaccount.

3. From the left-side navigation, choose **Applications > Subscriptions**.
4. In the *HTML5 Applications* table, choose the **Unsubscribe** button for the **ips** application.
5. Repeat the same in the *Java Applications* table for all Identity Provisioning applications (**ips**, **ipsproxy**, and **idds**).

#### Note

If the **Unsubscribe** button is missing for these Java applications, please create an incident to component **BC-IAM-IPS** and we'll remove the subscriptions for you.

**Next Steps:** You can now delete your Identity Provisioning subaccounts or keep them for future use.

#### Remember

If you want to permanently stop working with the Identity Provisioning, you have to terminate your contract with the service.

## 1.13 Submitting Improvement Requests

You can submit improvement requests for Identity Provisioning to the SAP Customer Influence site, a central place for all product improvement requests.

SAP Customer Influence provides an easy way to suggest ideas and request improvements. You can submit improvement requests, browse all of them, comment, vote, receive updates, and see who has voted for them.

#### Note

Before you submit your improvement, check if a similar improvement hasn't already been submitted. If a similar improvement is already in the system, vote for it instead of submitting a new improvement request.

Proceed as follows:

1. Access the SAP Customer Influence site at <https://influence.sap.com/sap/ino/#/campaign/2277>.
2. Log in with your S-user ID.
3. Choose *Improvement Request* on the left panel.
4. Enter the details for your improvement request. You can also provide links and attachments or add co-authors.

Field	Description
<i>Project</i>	The name of the current product session. Don't change it.
<i>Title</i>	Enter a title that clearly states your request and area of improvement.
<i>Category</i>	Select <i>Security Services</i> from the drop-down list.





Field	Description
<a href="#">Description</a>	Describe your idea. Provide as much details as possible.
<a href="#">Tags</a>	(Optional) Add tags that can help others find your request.

5. Choose [Submit](#).


## 1.14 Getting Support

If you have questions or encounter an issue while working with the Identity Provisioning service, you can address them through the communication channels listed below.


Use the following support media:

- [SAP Community: Ask a question](#) 
- [Identity Provisioning: Guided Answers](#) 
- [Identity Provisioning: Knowledge Base Articles](#) 
- [SAP Support Portal](#)  (An S-user is required to sign in and create an incident.)


### How to view my Identity Provisioning tenants?

As an SAP customer, you can view all your Identity Authentication and Identity Provisioning tenants by accessing the [SAP Cloud Identity Services - Tenants](#) application at the following URL: <https://iamtenants.accounts.cloud.sap/> . It displays the type of the tenant (test or productive), the date it was created, the region where it is available and the tenant administrators.

For more information about the data you can view and how to log on, see:

- [Viewing Assigned Tenants and Administrators](#)
- [New! Check on one single page all of your Identity Authentication and Identity Provisioning tenants and administrators](#) 

### How to create an incident?

1. Sign in to [SAP Support Portal](#) .
2. Choose [Report an Incident](#). *SAP ONE Support Launchpad* opens.
3. Perform a search to check whether a similar incident has already been reported.
4. If you cannot find any relevant incidents, create your own.
5. For [Component](#), enter: **BC-IAM-IPS**
6. Fill in the mandatory fields.

7. Explain your problem. We recommend including the following information in the incident:
  - Specify if your Identity Provisioning is purchased as a separate solution, or is part of a SAP cloud product license.
  - Region information
    - If you have purchased the Identity Provisioning before **September 1, 2020**, you can access the service in all Neo regions. To check the full list, see:  
[SAP BTP Discovery Center: Identity Provisioning](#)
    - If you have purchased the Identity Provisioning after **September 1, 2020**, you can access the service in all regions and data centers where the Identity Authentication is supported. To check the full list, see:  
[SAP BTP Discovery Center: Identity Authentication](#)
  - Subaccount technical name
  - The URL to the Identity Provisioning administration console, where the incident or error has occurred
  - The steps or clicks used to replicate the error
  - Screenshots, videos, or changed transformation mappings

## How to ask a question in SAP Community?

1. Sign in to [Identity Provisioning: Questions](#).
2. Choose [Ask a Question](#).
3. Enter the short and full text of your question or feedback.
4. **Identity Provisioning** is selected as a primary tag. If you need more tags, use the search to add them.
5. Choose [Submit your question](#).
6. A page dedicated to your feedback is created. On this page, you can check for answers from SAP developers and other users.
7. If you want to receive e-mail notifications from this page, choose [Follow](#).

## Account Information

### Caution

This section is not relevant for tenants created with Identity Provisioning purchases made after September 1, 2020.

On section [Support](#) in the Identity Provisioning user interface, you can see the SAP Business Technology Platform information relevant to your tenant (region host, global account, and subaccount name).

There is a UI message that informs you on how many subaccounts you have enabled the Identity Provisioning service for your global account. The default maximum number is **2** – one subaccount you can use for [testing](#) purposes, and one for [productive](#) scenarios. For more information, see [Access Identity Provisioning UI of Standalone Tenants \[page 1614\]](#) → first procedure → **step 2**.

If you have reached the maximum number of enabled subaccounts, a warning message will appear. If this number is insufficient to your business needs, you can raise the quota for your global account. Request more subaccounts by creating an incident for component **BC-IAM-IPS**.

## Related Information



[Monitoring and Troubleshooting \[page 1593\]](#)

# Important Disclaimers and Legal Information

## Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.

About the icons:

- Links with the icon  : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon  : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

## Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

## Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

## Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

## Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

© 2023 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see <https://www.sap.com/about/legal/trademark.html> for additional trademark information and notices.