



PUBLIC

Document Version: 2210 – 2022-10-11

API Tutorial for SAP Asset Intelligence Network

Content

- 1 Introduction. 4**
- 2 Rate Limiting and Throttling. 7**
- 3 Announcements. 14**
 - 3.1 Additional Information. 15
- 4 Company Profile. 19**
- 5 Composite API. 21**
- 6 Mass API. 35**
- 7 Documents. 39**
 - 7.1 Additional Information. 42
- 8 Equipment. 48**
 - 8.1 Equipment Search. 55
 - 8.2 Creating a Baseline. 62
 - 8.3 View all the Baselines Created for an Equipment. 63
 - 8.4 Editing a Baseline Description. 63
 - 8.5 View a Specific Baseline. 64
 - 8.6 Retrieve Header Information for a Baselined Equipment. 65
 - 8.7 View Components Information. 65
 - 8.8 View Template Information. 66
 - 8.9 View Values Information. 66
 - 8.10 View Spare Parts Information. 67
 - 8.11 View Documents Information. 68
 - 8.12 View a Specific Async Operation. 68
 - 8.13 View all Async Operations. 69
- 9 Equipment Requests. 70**
- 10 External IDs. 72**
- 11 Equipment Sync. 74**
- 12 Failure Modes. 75**
- 13 Functions API. 79**
- 14 Functional Locations. 80**

15	Groups	84
15.1	Additional Information	85
16	Improvement Requests	87
16.1	Additional Information	88
17	Instructions	93
17.1	Additional Information	94
18	Locations	102
19	Models	104
19.1	Model Search	108
20	Model Requests	115
21	Network Authorizations	117
22	Notifications	119
23	Organizational Structure	126
24	Spare Parts	127
25	Systems	129
25.1	System Topology	130
26	Templates	135
26.1	Additional Information	139
27	Unit of Measure Conversion	146
28	User Authorization	147
29	Work Orders	149
30	Searching Objects in SAP Asset Intelligence Network	150
31	OData Services for SAP Analytics Cloud	151
31.1	Reading Data of Equipment	151
31.2	Reading Data of Indicators	152
31.3	Reading Data of Models	152
31.4	Reading Data of Locations	153
31.5	Reading Data of Spare Parts	153
31.6	Reading Data of Notifications	154
31.7	Reading Data of Work Orders	154
32	Asynchronous Operations	155

1 Introduction

The API tutorial guides you in using APIs provided by the SAP Asset Intelligence Network.

i Note

With the movement of asset central foundation APIs to SAP API hub, you can see that some of the API links will redirect to the API hub while some of the links redirect to the HTML files.

API	Link
Announcement API	https://api.sap.com/api/ManageAnnouncementsAPI/resource
Baseline API	https://help.sap.com/doc/364912daaa744a448e846f78a540e162/2105/en-US/baseline.html
Equipment API	https://api.sap.com/api/EquipmentAPI/resource
Model API	https://api.sap.com/api/ModelAPI/resource
Notifications API	https://api.sap.com/api/EquipmentAPI/resource
Organizational Structure API	https://api.sap.com/api/OrganizationalStructure/resource
Systems API	https://api.sap.com/api/Systems/resource
Company Profile API	https://api.sap.com/api/CompanyProfileAPI/resource
Composite and Mass API	https://api.sap.com/api/CompositeAndMassAPI/resource
Document API	https://api.sap.com/api/DocumentAPI/resource
Equipment Request API	https://api.sap.com/api/EquipmentRequestAPI/resource
Equipment Sync API	https://api.sap.com/api/EquipmentSyncAPI/overview
External ID API	https://api.sap.com/api/ExternalIDsAPI/resource
Failure Modes API	https://api.sap.com/api/FailureModesAPI/resource
Functions API	https://api.sap.com/api/FunctionAPI/resource
Functional Locations API	https://api.sap.com/api/FunctionalLocationAPI/resource
Groups API	https://api.sap.com/api/GroupsAPI/resource
Instructions API	https://api.sap.com/api/Instructions/resource
Location API	https://api.sap.com/api/LocationAPI/resource
Model Request API	https://api.sap.com/api/ModelRequestAPI/resource

API	Link
Network Authorization API	https://api.sap.com/api/NetworkAuthorizationAPI/resource
Improvement Request API	https://api.sap.com/api/ImprovementRequestAPI/resource
Spare Parts API	https://api.sap.com/api/SparePartsAPI/resource
Template API	https://api.sap.com/api/TemplateAPI/resource
Unit of Measures API	https://api.sap.com/api/UnitOfMeasuresAPI/resource
User Authorizations API	https://api.sap.com/api/UserAuthorizations/resource
Work Orders API	https://api.sap.com/api/WorkOrdersAPI/resource
Asynchronous Operations API	https://help.sap.com/doc/971fafd93d864a2098a63afd6b5d2f5d/2111/en-US/AsyncOps.html

APIs in the SAP API Hub

API package details:

Level	Name
API package	Collection of related APIs, belonging to one product, for example, SAP Asset Intelligence Network
APIs	Collection of REST operations available for each API.
API References	A remote data entity, identified by a URI, on which operations are performed.
Operation	Data operation, such as GET, POST, PUT, UPDATE, or DELETE.

Each API reference provides the following information:

- What the API does
- URLs of the HTTP requests
- Description of the request path, query, and body parameters
- Description of the response content

Test these APIs directly in the [SAP API Business Hub](#).

The parameters for the operations are detailed in the *Model* tab under **Parameters** > *body*.

Steps to Fetch Application URL- For SAP Cloud Platform - Neo Environment

Prerequisite

You must have access to SAP Cloud Platform cockpit for your account.

Steps

1. Log in to SAP Cloud Platform cockpit and navigate to your Global Account.
2. Select your *Subaccount* in the *Global Account*.
3. In your *Subaccount* navigate to **Applications** > **Subscriptions**.
4. In *Subscriptions* tab, you will find table of **Subscribed Java Applications**.
5. Click on **ain** application from the table.
6. In *Overview* tab, you will find the **Application URLs** section.

Application URL will be of the format **https://ain<>-<>.hana.ondemand.com/ain**.

Steps to Fetch Application URL- For SAP Cloud Platform - Cloud Foundry Environment

You can find the application URL from the service key instance using navigation path Subaccount → Spaces → Service Instances → (select relevant service instance) → Service Keys → (select relevant service key). you can find Client Credentials in the service key-in the following parameters:

Client Credentials	Parameters
Client ID	uaa.clientid
Client Secret	uaa.clientsecret
Token URL	uaa.url + '/oauth/token'
Application URL	endpoints.ain-service

Related Information

[Accessing APIs of SAP Intelligent Asset Management Solutions](#)

2 Rate Limiting and Throttling

This section provides information on the rate limiting and API throttling in SAP Asset Intelligence Network.

Throttling

Throttling places restrictions on the frequency of the requests made by the client and is a mechanism designed to protect SAP Asset Intelligence Network from increased traffic into public APIs. These restrictions will not impact user consumption through standard UIs (user interfaces) delivered by SAP.

To prevent APIs from being overloaded by too many requests, we have introduced API throttling to regulate the API usage per tenant. The use by client applications or background job threads that make multiple high volume calls on a continuous basis in order to extract or import data, or update can have unintended consequences and impact underlying **resource availability**, cause **performance issues**, and ultimately interrupt customer operations.

Rate Limiting

The table below lists the proposed limits for the APIs and the limits will be applicable from 2111 release. The list of APIs will be extended in the future releases. (watch this space for any further updates)

List of APIs and their Limits

Module	API	Method	Functionality	Replenish Rate (request/second)	Applicable From Release
Equipment	/services/api/v1/equipment({id})/components	GET	Get Equipment Components	1	2205
Equipment	/services/api/v1/equipment({id})/components	PUT	Update Equipment Components	1	2205
Equipment	/services/api/v1/equipment({id})/components	PATCH	Update Equipment Components	1	2205
Equipment	/services/api/v1/equipment({id})/values	GET	Get Equipment Attribute Values	1	2205
Equipment	/services/api/v1/equipment({id})/values	PUT	Update Equipment Attribute Values	1	2205

Module	API	Method	Functionality	Replenish Rate (request/second)	Applicable From Release
Equipment	/services/api/v1/ equipment({id})/ values	PATCH	Update Equipment Attribute Values	1	2205
Model	/services/api/v1/ models({id})/ components	GET	Get Model Compo- nents	1	2205
Model	/services/api/v1/ models({id})/ components	PUT	Update Model Components	1	2205
Model	/services/api/v1/ models({id})/ components	PATCH	Update Model Components	1	2205
Model	/services/api/v1/ models({id})/ values	GET	GET Model Attrib- ute Values	1	2205
Model	/services/api/v1/ models({id})/ values	PUT	Update Model At- tribute Values	1	2205
Model	/services/api/v1/ models({id})/ values	PATCH	Update Model At- tribute Values	1	2205
Systems	/services/api/v1/ systems({id})/ components	GET	Get System Com- ponents	1	2205
Systems	/services/api/v1/ systems({id})/ components	PUT	Update System Components	1	2205
Systems	/services/api/v1/ systems({id})/ components	PATCH	Update System Components	1	2205
Systems	/services/api/v1/ systems({id})/ values	GET	Get System Attrib- ute Values	1	2205
Systems	/services/api/v1/ systems({id})/ values	PUT	Update System At- tribute Values	1	2205
Functional Loca- tions	/services/api/v1/ floc({id})/compo- nents	GET	Get Functional Lo- cation Compo- nents	1	2205
Functional Loca- tions	/services/api/v1/ floc({id})/compo- nents	PUT	Update Functional Location Compo- nents	1	2205
Functional Loca- tions	/services/api/v1/ floc({id})/compo- nents	PATCH	Update Functional Location Compo- nents	1	2205

Module	API	Method	Functionality	Replenish Rate (request/second)	Applicable From Release
Functional Locations	/services/api/v1/floc({id})/values	GET	Get Functional Location Attribute Values	1	2205
Functional Locations	/services/api/v1/floc({id})/values	PUT	Update Functional Location Attribute Values	1	2205
Spare Parts	/services/api/v1/parts({id})/values	GET	Get Parts Attribute Values	1	2205
Spare Parts	/services/api/v1/parts({id})/values	PUT	Update Parts Attribute Values	1	2205
Documents	/services/api/v1/documents/assign/equipment	POST	Assign document to equipment	5	2111
Documents	/services/api/v1/equipment({equipmentId})/documents	GET	Get documents assigned to equipment	5	2111
Documents	/services/api/v1/documents	POST	Upload document	1	2111
Documents	/services/api/v1/documents	GET	List documents	1	2111
External System/IDs	/services/api/v1/object/externalids	POST	Maintain external IDs	2	2111
External System/IDs	/services/api/v1/object/externalid	PUT	Update external ID	2	2111
External System/IDs	/services/api/v1/objectid/ainobjects({externalID})	GET	Get all internal IDs for external ID	5	2111
External System/IDs	/services/api/v1/objectid/externalid({objectID})	GET	Get All external IDs for internal ID	5	2111
External System/IDs	/services/api/v1/external/systems	GET	Get all external systems	5	2111
Indicator	/services/api/v1/equipment({id})/indicatorvalues	GET	Get indicator values for equipment	2	2111
Indicator	/services/api/v1/floc({id})/indicatorvalues	GET	Get indicator values for functional location	2	2207
Indicator	/services/api/v1/indicators	GET	List Indicators	1	2111
Indicator	/services/api/v1/indicatorvalues	GET	Get Indicator values of particular indicator	2	2111

Module	API	Method	Functionality	Replenish Rate (request/second)	Applicable From Release
Location	/services/api/v1/location({id})	GET	Get location header	5	2111
Location	/services/api/v1/location	GET	List Locations	1	2111
Location	/services/api/v1/location({id}/components	GET	Get location components	2	2111
Location	/services/api/v1/location({id}/components	PUT	Update location components	1	2111
Location	/services/api/v1/location({id}/equipment	PATCH	Update equipment of a location	1	2111
Location	/services/api/v1/location({id}/values	PUT	Update attributes of a location	2	2111
Location	/services/api/v1/location({id}/values	GET	Retrieve attributes of a location	2	2111
Location	/services/api/v1/location({id}/attachments	GET	Retrieves documents for a location	2	2111
Equipment	/services/api/v1/equipment	GET	List Equipment	1	2111
Notification	/services/api/v1/notification	POST	Create Notification	1	2111
Notification	/services/api/v1/notification	GET	List Notifications		2111
Workorder	/services/api/v1/workorder	POST	Create Work Order	1	2111
Workorder	/services/api/v1/workorders	GET	List Workorders		2111
Templates	/services/api/v1/attributes	POST	Create template Attributes	1	2111
Templates	/services/api/v1/attributes	GET	List attributes	1	2111
Templates	/services/api/v1/attributes/{attributeID}	GET	Get Attribute by Id	2	2111
Templates	/services/api/v1/attributes/{attributeID}/equipment	GET	Retrieves the equipment's where this attribute has a value	1	2111

Module	API	Method	Functionality	Replenish Rate (request/second)	Applicable From Release
Templates	/services/api/v1/attributes/{attributeID}	PUT	Updates an attribute	2	2111
Templates	/services/api/v1/attributegroups	GET	List Attribute Group	1	2111
Templates	/services/api/v1/codelists	GET	List codelists	1	2111
Templates	/services/api/v1/codelists/{codelistID}	GET	Retrieve a codelist	2	2111
Templates	/services/api/v1/codelists/{codelistID}	PUT	Update a codelist	2	2111
Templates	/services/api/v1/templates	GET	Retrieves templates	1	2111
Templates	/services/api/v1/templates	POST	Create Templates	2	2111
Templates	/services/api/v1/templates/\$count	GET	Retrieves the count of templates	1	2111
Templates	/services/api/v1/templates/{templateID}	GET	Retrieve a template	3	2111
Templates	/services/api/v1/templates/{templateID}	PUT	Edit template	2	2111
Templates	/services/api/v1/nested/structures	GET	List Nested structures	1	2111
Templates	/services/api/v1/nested/structures	POST	Create nested Structures	1	2111
Templates	/services/api/v1/nested/structures/{nested-StructureID}	GET	Retrieve a nested structure	1	2111
Templates	/services/api/v1/nested/structures/{nested-StructureID}	PUT	Edit a nested structure	1	2111
Spare Parts	/services/api/v1/parts	GET	Retrieves list of spare part	1	2111
Spare Parts	/services/api/v1/parts	POST	Create a spare part	2	2111
Spare Parts	/services/api/v1/parts	PUT	Updates details of specified spare part	2	2111

Module	API	Method	Functionality	Replenish Rate (request/second)	Applicable From Release
Spare Parts	/services/api/v1/parts({partid})	GET	Retrieves spare part header information for specific spare part	3	2111
Spare Parts	/services/api/v1/parts({partid})	PUT	Updates spare part details	1	2111
System	/services/api/v1/systems	GET	List Systems	1	2111
Model	/services/api/v1/models	GET	List Models	1	2111
Functional Location	/services/api/v1/floc	GET	List Functional Locations	1	2111
Composite API	/ac/services/api/v1/composites or /services/api/v1/composites	POST	Composite API	50	2111

Note

This is the initial list of APIs and more will be added in a phased manner. The API header will provide the details of the limits as explained below.

- The limits are currently expressed in terms of per second, but we intend to extend the limits at multiple frequencies per **second/minute/hour/day** in the upcoming releases.
- Limits are specific to tenant type, that is, dev/test tenant type has lower limits when compared with production tenant type.

In case the set limit is exceeded, API calls result in **HTTP 429** error. You can retry after the set time interval and also see if you can spread calls according to limits so that **HTTP 429** errors can be avoided.

Replenish rate: Number of requests allows in 1 second.

Example

- Sample 429 response
HTTP status code : 429
Response body: <empty>
- Headers
Each API call header has the following details, current limit information, consumed status, and so on.
 - Tenant Level limit
 - x-ratelimit-replenish-rate
Indicates how many queries are permitted per second
 - x-ratelimit-remaining
Indicates how many queries you have left in the current second

i Note

Absence of headers indicates that rate limit is not enabled for tenant or API.

i Recommendation

In case you are building custom solution/integration that needs you to utilize too many API calls, we advise you to seek guidance from the SAP Asset Intelligence Network team by creating a ticket (Component: SCM-AIN-APL). This helps us to understand your needs better and also suggest best way to achieve your business needs.

3 Announcements

An announcement is the information that a manufacturer communicates to an operator about an asset. For example, a change in an existing instruction, or an update to the spare parts of an item of equipment. SAP Asset Intelligence Network allows you to create the following types of instructions:

- CDO:/content/authoring/xax1592895445199.xml
- **Instruction change:** An announcement type communicated by the manufacturer when an instruction related to a model undergoes a change.
- **Service Bulletin:** An announcement type communicated by the manufacturer to indicate a change to the model due to safety, performance, or statutory reasons. The change is carried out by the operator. Example: change a part that has caused.
- **Recall:** An announcement type communicated by the manufacturer to indicate a change to the model due to safety, performance, or statutory reasons. The change is carried out by the manufacturer.
- **New Policy:** An announcement type communicated by the manufacturer that specifies a recommendation, or a guidance by the manufacturer to use an item of equipment or a model within a company.
- **New Model:** An announcement type communicated by the manufacturer to communicate availability of new, or improved models.
- **Attachment Change:** An announcement type communicated by the manufacturer to communicate when any documents, or drawings related to a model undergoes a change.
- **Spare Parts Change:** An announcement type communicated by the manufacturer when any spare parts related to the model undergoes a change. Example: manufacturer part number changed.
- **Model Information Change:** An announcement type communicated by the manufacturer when a model undergoes a change in its specification, or any relevant metadata of a model.

For simplicity, the following sections deal with an Instruction change announcement.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles ANNOUNCEMENT_READ or ANNOUNCEMENT_EDIT or ANNOUNCEMENT_DELETE assigned to your user ID in the SAP Cloud Platform account.
- **API hub link:** <https://api.sap.com/api/ManageAnnouncementsAPI/resource>

Methods

Group	Method	Action	URI
Announcement	GET	Retrieves list of announcements for a manufacturer	Application_URL/services/api/v1/announcements
	POST	Creates an announcement	Application_URL/services/api/v1/announcements

Group	Method	Action	URI
	PUT	Update an announcement	Application_URL/ services/api/v1/announcements
	GET	Retrieves the count of announcements	Application_URL/ services/api/v1/announcements/\$count
	POST	Delete announcements according to announcement IDs	Application_URL/ services/api/v1/announcements/delete
	PUT	Publish announcements	Application_URL/ services/api/v1/announcements/publish
	GET	Retrieves announcements with announcement IDs	Application_URL/ services/api/v1/announcements({id})
Models	GET	Retrieves list of models assigned to announcement ID	Application_URL/ services/api/v1/announcements({id})/models

3.1 Additional Information

The following table provides information about the fields of the request payload, along with the corresponding data types, and the data type descriptions:

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
announcementName (Mandatory)	string	Specifies the name of the announcement.
announcementDescription (Optional)	string	Specifies the description for the announcement.
announcementTypeCode (Mandatory)	number	Specifies the type code for the type of an announcement. For more information on announcement type codes, see the Announcement Type Codes table.
source (Optional)	String	Specifies the name of the business partner who created the announcement.

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
priorityCode (Mandatory)	Number	Specifies the importance of the announcement. The following list contains the possible values and their descriptions: <ul style="list-style-type: none"> • 1- Low • 2- Medium • 3 - High
publishedDate (Optional)	Date	Specifies the date when the announcement was published.
statusCode (Mandatory)	Number	Specifies the status of the announcement. The following list contains the possible values and their descriptions: <ul style="list-style-type: none"> • 1- Unpublished • 2 - Published • 3 - In Revision
modelIDs (Optional)	Array	Specifies the list of model IDs that you can assign to an announcement.
modelID (Optional)	String	Specifies the model ID that you can assign to an announcement. <div data-bbox="1007 1216 1398 1386" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>i Note</p> <p>The model ID is a unique ID generated by the system when you create a model. (see Creating a Model)</p> </div>
documentIDs (Optional)	Array	Specifies the list of document IDs that you can assign to an announcement.
documentID (Optional)	String	Specifies the document ID that you can assign to an announcement. <div data-bbox="1007 1601 1398 1800" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>i Note</p> <p>The document ID is a unique ID generated by the system when you upload a document. (see Uploading an Attachment)</p> </div>
instructionIDs (Optional)	Array	Specifies the list of instruction IDs that you can assign to an announcement.

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
instructionID (Optional)	String	Specifies the instruction ID that you can assign to an announcement. <div style="border: 1px solid #ccc; background-color: #f9f9f9; padding: 5px;"> <p>i Note</p> <p>The instruction ID is a unique ID generated by the system when you create an instruction. (see Creating an Instruction)</p> </div>
assignUnassignFlag (Optional)	String	A flag that specifies if you can assign a model, instruction, or a document. Possible values include 'a' for assign and 'ua' for unassign.
isDefault (Optional)	String	A flag that specifies if the assigned instruction to the announcement is a default one. Possible values: 1- default 0 - not default <div style="border: 1px solid #ccc; background-color: #f9f9f9; padding: 5px;"> <p>i Note</p> <p>Additionally, this flag is set to 1 when you create an announcement using the Instructions app.</p> </div>

Announcement Type Codes

Announcement Type Code	Description
1	Instruction Change
2	Service Bulletin
3	Recall
4	New Policy
5	New Model
6	Attachment Change
7	Spare Parts Change
8	Model Information Change


The following table provides information related to the fields of the response payload along with the corresponding data types, and descriptions. To avoid repetition, fields that are common with the request payload are excluded in the following table:

Field Name	Data Type	Description and Possible Values
title, name	String	Specifies the name or a title of the document.
phase	String	Specifies the phase in the lifecycle of a model or an item of equipment that the document is relevant for. For more information on possible values, see the Phase Codes and Category Codes table.
category	String	Specifies the category for a phase in the lifecycle of a model, or an item of equipment that the document is relevant for. For more information on possible values, see the Phase Codes and Category Codes table.
fileSize	Integer	Specifies the size of a document.
fileSizeUnit	Integer	Specifies the unit used to measure the file size.
mimeType	String	Specifies the mimetype of the document.

4 Company Profile

You can view the details of an organization, its locations, invitees and subsidiaries that are in SAP Asset Intelligence Network.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles COMPANYPROFILE_READ or COMPANYPROFILE_EDIT or COMPANYPROFILE_DELETE assigned to your user ID. Also, for user APIs you need CONFIGURATION_EDIT or CONFIGURATION_DELETE assigned to your user ID.
- **API hub link:** [Company Profile API](#) 

Methods

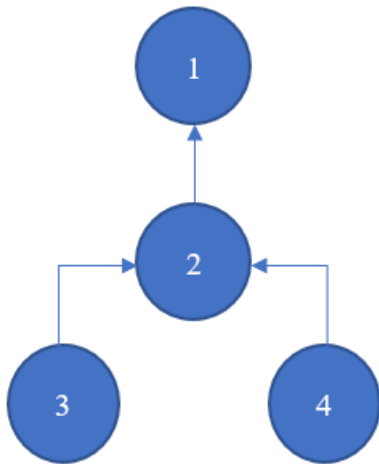
Group	Method	Actions	URI
Organization	GET	Returns company profile information	Application_URL/services/api/v1/company/profile({orgId})
	GET	Returns company profile location	Application_URL/services/api/v1/company/profile({orgId})/locations
	GET	Returns count of company profile locations	Application_URL/services/api/v1/company/profile({orgId})/locations/\$count
	GET	Returns subsidiaries information	Application_URL/services/api/v1/company/profile({orgId})/subsidiaries
	GET	Returns subsidiaries count	Application_URL/services/api/v1/company/profile({orgId})/subsidiaries/\$count
Invitees	GET	Returns the invitees company profile	Application_URL/services/api/v1/company/profile/invitees
	POST	Creates the invitee company profile	Application_URL/services/api/v1/company/profile/invitees
	GET	Returns invitees company profile count	Application_URL/services/api/v1/company/profile/invitees/\$count

Group	Method	Actions	URI
Subsidiaries	POST	Creates subsidiary for the organizations or invitee	Application_URL/ services/api/v1/company/ profile/subsidiaries
Users	GET	Returns users	Application_URL/ services/api/v1/company/ users
	PUT	Updates user details	Application_URL/ services/api/v1/company/ users
	POST	Creates user	Application_URL/ services/api/v1/company/ users
	DELETE	Deletes a user	Application_URL/ services/api/v1/company/ users({businessPartnerId})
Others	GET	Returns enumerations	Application_URL/ services/api/v1/company/ profile/enumerations
	GET	Returns organizations based on roles	Application_URL/ services/api/v1/organiza- tions/byrole
External Organizations	GET	Returns a list of the associated external organizations	Application_URL/ services/api/v1/company/ profile/externalorg
	POST	Creates an external organization	Application_URL/ services/api/v1/company/ profile/externalorg
	PUT	Update an external organization	Application_URL/ services/api/v1/company/ profile/externalorg
	GET	Returns external organizations count	Application_URL/ services/api/v1/company/ profile/externalorg/\$count
	DELETE	Deletes an external organization	Application_URL/ services/api/v1/company/ profile/externalorg/{externalOrgId}
	PUT	Delete external organizations	Application_URL/ services/api/v1/company/ profile/externalorgs

5 Composite API

Composite API contains list of API calls, to be executed depending on the predefined dependency.

The independent API call is executed first and later all the subsequent calls, which depend on this API call's response properties is to be substituted either in the URL or the payload and is dependents are executed in sequence.



Composite API - Graphical Representation

Consider the following example (based on the above graphical representation):

Request ID	HTTP Method	API	Comments	Request Payload	Response JSON
1	POST	/ain/services/api/v1/models	Create model 'Sub Sea Pump' in asset central foundation.	<pre> { "internalId": "Sub Sea Pump", "templates": [{ "id": "templa te-id", "primary": true }], "organizationID": "OrgId" , ... } </pre>	<pre> { "name": "Sub Sea Pump", "internalId": " Sub Sea Pump", "organizationID": "orgId" , ... "modelId": "modelId", ... } </pre>

Request ID	HTTP Method	API	Comments	Request Payload	Response JSON
2	PUT	/ain/ services/api/v1/ models({@1.re- sponse.modelId})/ publish	Publish model 'Sub Sea Pump' in asset central foun- dation		<pre> { "name": "Sub Sea Pump", "intern alId": " Sub Sea Pump", "organi zationI D": "orgId" , ... "modelI d": "modelI d", ... } </pre>

Request ID	HTTP Method	API	Comments	Request Payload	Response JSON
3	POST	/ain/services/api/v1/equipment	Create equipment 'Pump1' in Asset central with the model 'Sub Sea Pump'	<pre> { "internalId": "Pump1", "modelId": "{@2.response.modelId}", "buildDate": "2019-03-16", "serialNumber": "R73FR9A", ... } </pre>	<pre> { "name": "Pump1", "internalId": "Pump1", "modelId": "modelId", "modelId": "modelId", ... } </pre>
4	POST	/ain/services/api/v1/equipment	Create equipment 'Pump2' in Asset central with the model 'Sub Sea Pump'	<pre> { "internalId": "Pump2", "modelId": "{@2.response.modelId}", "buildDate": "2019-03-16", ... } </pre>	<pre> { "name": "Pump2", "internalId": "Pump2", "modelId": "modelId", "modelId": "modelId", ... } </pre>

Above request can be composed together as composite request as below:

Request ID	HTTP method	API URL	Comments	Request Payload	Response JSON
UUID	POST	/ac/services/api/v1/composites	Create composite request.	<pre> { "entities": [{ "id": "1", "request": { "url": "{{ac-url}}/action/service/api/v1/models", "method": "POST", "payload": { "internalId": "SubSeaPump", "templates": [{ "id": "template-id", </pre>	<pre> { "id": "8a78c762-81e2-49bf-80cb-b927bf6c2207", "statusCode": 200, "entities": [{ "id": "1", "request": { "url": "{{ac-url}}/action/service/api/v1/models", "method": "POST", "payload": { "internalId": "SubSeaPump", </pre>

Request ID	HTTP method	API URL	Comments	Request Payload	Response JSON
				<pre> "primary": true }], "organizationI D": "OrgId" , .. } } }, { "id": "2", "request": { "url": "{{ac- url}}/a in/ service s/api/v 1/ models({@1.res ponse.m odelId})/ publish ", "method ": "PUT" </pre>	<pre> "templa tes": ["id": "templa te-id", "primary": true }], "organi zationI D": "OrgId" , .. } } }, { "id": .. } "respon se": { "status Code": 200, "body": { "name": "Sub Sea Pump", </pre>

Request ID	HTTP method	API URL	Comments	Request Payload	Response JSON
				<pre> } }, { "id": "3", "request": { "url": "{{ac- url}}/a in/ service s/api/v 1/ equipme nt", "method ": "POST", "payload": { "intern alId": "Pump1" , "modelId": "{@2.re sponse. modelId }", "buildD ate": "2019-0 3-16", } } }], "intern alId": "Sea Pump", "organiza tionID": "orgId" , .. "modelId": "modelId", .. } } , { "id": "2", "request": { "url": "{{ac- url}}/a in/ service s/api/v 1/ models({@4- createM odel.re sponse. </pre>	<pre> "intern alId": " Sea Pump", "organiza tionID": "orgId" , .. "modelId": "modelId", .. } } , { "id": "2", "request": { "url": "{{ac- url}}/a in/ service s/api/v 1/ models({@4- createM odel.re sponse. </pre>

Request ID	HTTP method	API URL	Comments	Request Payload	Response JSON
				<pre> .. } } }, { "method": "PUT", "response": { "statusCode": 200, "body": { "name": "Sub Sea Pump", "internalId": "Sub Sea Pump", "payload": { "organizationId": "Pump2", "orgId": "Pump2", "modelId": "Pump2", "modelId": "Pump2", "buildDate": "Pump2" } } } </pre>	<pre> modelId })/ publish ", "method": "PUT", "response": { "statusCode": 200, "body": { "name": "Sub Sea Pump", "internalId": "Sub Sea Pump", "payload": { "organizationId": "Pump2", "orgId": "Pump2", "modelId": "Pump2", "modelId": "Pump2", "buildDate": "Pump2" } } } </pre>

Request ID	HTTP method	API URL	Comments	Request Payload	Response JSON
				<pre>"2019-03-16", .. } } }]</pre>	<pre>. } } }, { "id": "3", "request": { "url": "{{ac-url}}/a in/ service s/api/v 1/ equipme nt", "method": "POST", "payload": { "internalId": "Pump1", "modelId": "@2.response.modelId", "buildDate":</pre>

Request ID	HTTP method	API URL	Comments	Request Payload	Response JSON
					<pre> "2019-0 3-16", "serial Number" : "R73FR9 A", .. } }, "respon se": { "status Code": 200, "body": { "name": "Pump1" , "intern alId": "Pump1" , "modelI d": "modelI d", .. } </pre>

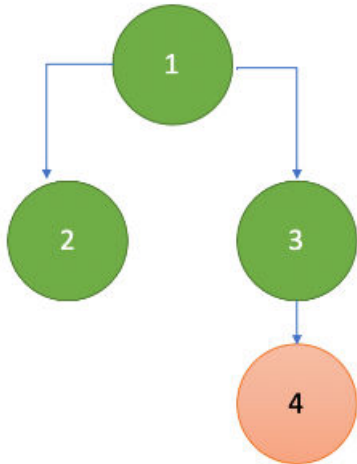
Request ID	HTTP method	API URL	Comments	Request Payload	Response JSON
					<pre> } }, { "id": "4", "request": { "url": "{{ac- url}}/a in/ service s/api/v 1/ equipme nt", "method": "POST", "payload": { "internalId": "Pump1" , "modelId": "{@2.re sponse. modelId }", "buildDate": "2019-0 3-16", </pre>

Request ID	HTTP method	API URL	Comments	Request Payload	Response JSON
					<pre> "serial Number" : "R73FR9 A", . .. } }, "respon se": { "status Code": 200, "body": { "name": "Pump1" , "intern alId": "Pump1" , "modelI d": "modelI d", . .. } } }] } </pre>

Request Entry

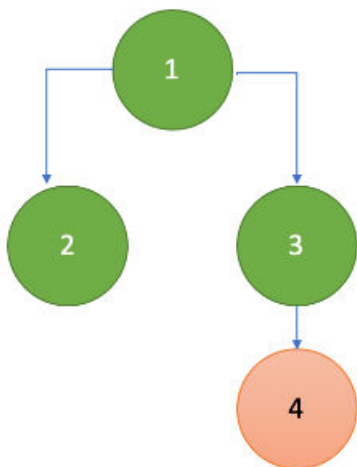
- **Option 1**

Composite request can be retried with the same requests as the original request, user can retry the composite with response body of the failed composite request, then composite is retried for the failed requests, this will be useful when there are some temporary errors with the downstream servers/network. Example: Considering the partial failure response provided below, when composite retried, succeeded requests '1', '2' and '3' will not be re-executed, but request '4' is failure, so only request '4' will be retried with same as original request.



- **Option 2**

Composite request can be retried with modifying requests in the original request, user has to provide the composite request payload as the response from the failed request and modify the payload accordingly. When the request is retried then composite execution is resumed from failure request and all the children also retried. Successful requests are not executed again, instead these responses are used from the provided value in the request. Modifying the request which is already success will not be re-executed. Example: Considering the partial failure response provided below, when composite retried, succeeded requests '1', '2' and '3' will not be re-executed, but request '4' is failure, so only request '4' will be retried with newly provided request information.



Placeholder Notation Conventions:

1. Field from the previous response
{@<prev-requestId>.response.<intended property from the response>}
2. Field from the previous response which is an array
{@<prev-requestId>.response[required-index].<intended property from the response>}

i Note

- Composite API supports up to 50 requests in single composite request.
- Make sure the provided token has necessary authorization for all the APIs.
- The content/type for every request is assumed to be **application/json**.
- Response status will be 200 if all the requests are successful(200-299 HTTP status codes), If one or more requests return non-success codes then status will be 207.
- Only error requests would be allowed to be edited when being retried. Successful POSTs would be ignored while retry.
- Ordering of dependencies not required, composite service will order the dependencies using the placeholder notations.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **API hub link:** [Composite and Mass API](#)

Methods

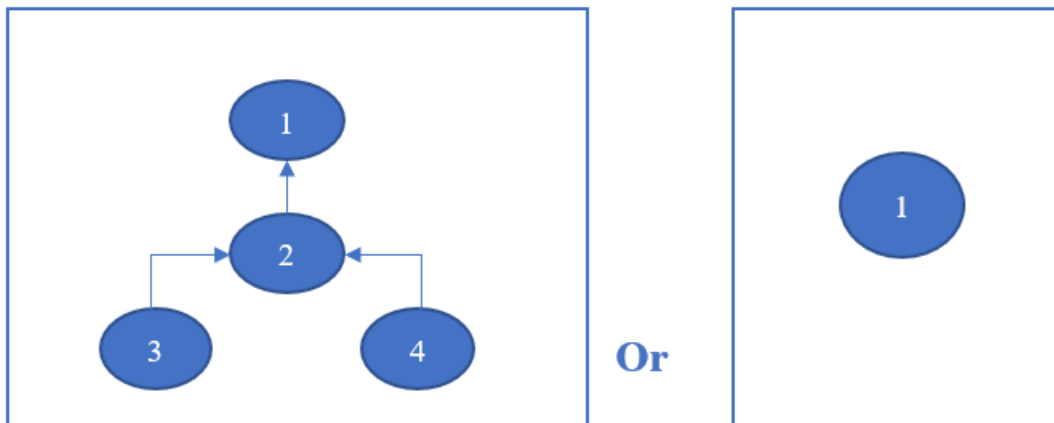
Group	Method	Actions	URI
Composite	POST	Create new composite request	Application_URL/services/api/v1/composites

i Note

The request payload must be in JSON format.

6 Mass API

Any API with a pattern in its end-point involving "mass" and an identifier of predefined length is used, it qualifies to run through the Mass API framework which is asynchronous in nature. The identifier serves as handler to later monitor the progress of the request as Mass is asynchronous in nature.



Mass API - Graphical representation

Mass Enabled API Graphical representation

- Mass API accepts a single request either with an AC API with request body if any or a composite request with payload.
- The mass API URL should be in the form of `"/ac/services/api/v1/mass/jobs/<jobid>/<actual url>"` where the `<actual url >` can be either the an AC API URL or Composite API URL. Use the same HTTP method as the actual URL.
- You can add a refresh token for mass API service to refresh the token automatically during the even the JWT token expiration. This is an optional header.

i Note

- In the dashboard the user will be able to monitor the jobs created by his/her own user. Technical user created jobs can be monitored by API only.
- If refresh token is not provided and token expired then request might fail.

Request ID	HTTP method	API	Comments	Request Payload	Header	Response
1	POST	<baseurl>/ac/services/api/v1/mass/jobs/<jobid>/api/v1/equipment	Create an equipment using mass API	<pre> { "internalId": "Test Equip Paolo Paolo 003", "modelId": "DEE350B5C5F24F058A2E4A39FA8B5B34", "sourceBPRole": "1", "modelKnown": true, "lifeCycle": "2", "descriptions": [{ "short": "TEST EQUIP PAOLO_003_Desc", "long": null, </pre>	<pre> "x-sap-refresh-token"=<refresh token> Authosrization: <JWT Token> </pre>	<pre> { id: <id> } </pre>

Request ID	HTTP method	API	Comments	Request Payload	Header	Response
				<pre> "language" : "en" }], "operatorID": "FFB5 88385 8C448 C9BA7 CC249 FAF44 988" } </pre>		

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **API hub link:** [Composite and Mass API](#)
The request payload must be in JSON format.

Methods


Group	Method	Actions	URI
Mass	POST	Create mass request for POST calls	Application_URL/ services/api/v1/mass/jobs/ {id}/{actual_url}
	PUT	Create mass request for PUT calls	Application_URL/ services/api/v1/mass/jobs/ {id}/{actual_url}
	DELETE	Create mass request for DELETE calls	Application_URL/ services/api/v1/mass/jobs/ {id}/{actual_url}
	GET	Create mass request for GET calls	Application_URL/ services/api/v1/mass/jobs/ {id}/{actual_url}
	GET	Get all jobs for the tenant/user	Application_URL/ services/api/v1/mass/jobs
	GET	Get the job details	Application_URL/ services/api/v1/mass/jobs/ {id}

Group	Method	Actions	URI
	GET	Get the mass request details	Application_URL/ services/api/v1/mass/jobs/ {jobId}/requests/{id}

7 Documents

A document contains information about a specific phases in the life cycle of a model, an item of equipment, an instruction, or an announcement.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles DOCUMENT_READ, DOCUMENT_EDIT, and DOCUMENT_DELETE assigned to your user ID.
- **API hub link:** [Document API](#) 

Methods

Group	Method	Actions	URI
List and Header	POST	Upload a document	Application_URL/ services/api/v1/documents
	DELETE	Deletes the specific documents	Application_URL/ services/api/v1/documents
	GET	Lists documents based on filter query parameter	Application_URL/ services/api/v1/documents
Document Files	GET	Download specific file	Application_URL/ services/api/v1/documents/ specificFile
	POST	Upload Logo	Application_URL/ services/api/v1/documents/ images
	POST	Upload file	Application_URL/ services/api/v1/documents/ {documentId}/language
	POST	Delete specific file	Application_URL/ services/api/v1/{documentId}/documentsDeleteFiles
	GET	Download Content of Document	Application_URL/ services/api/v1/documents/ {documentID}
	PUT	Update content of the document	Application_URL/ services/api/v1/documents/ {documentId}/language/ {language}

Group	Method	Actions	URI
	POST	Upload file descriptions	Application_URL/ services/api/v1/document/ upload/FileDescription
	PUT	Update file descriptions	Application_URL/ services/api/v1/document/ upload/FileDescription
Document Information	GET	Get all files of document	Application_URL/ services/api/v1/documents- File/{documentID}
	GET	Retrieves objects for this document is assign	Application_URL/ services/api/v1/documents/ {documentID}/assignedOb- jects
	GET	Information related to phase- Code	Application_URL/ services/api/v1/documents/ information
	PUT	Update phaseCode and cate- goryCode	Application_URL/ services/api/v1/documents/ information
	GET	Get all file extensions	Application_URL/ services/api/v1/document/ information/fileextensions
VDS document	POST	Upload VDS document	Application_URL/ services/api/v1/ documents/vds
	PUT	Update VDS document	Application_URL/ services/api/v1/ documents/vds/{vdsDocu- mentId}
Document Unassign	POST	Remove a model	Application_URL/ services/api/v1/documents/ unassign/model/docu- ments/unassign/model
	POST	Remove announcement	Application_URL/ services/api/v1/documents/ unassign/announcement
	POST	Remove steps	Application_URL/ services/api/v1/documents/ unassign/steps
	POST	Remove location	Application_URL/ services/api/v1/documents/ unassign/functionallocation
	POST	Remove functional location	Application_URL/ services/api/v1/documents/ unassign/floc

Group	Method	Actions	URI
	POST	Remove improvement case	Application_URL/ services/api/v1/documents/ unassign/improvementcase
	POST	Remove model request	Application_URL/ services/api/v1/documents/ unassign/modelrequest
	POST	Remove system	Application_URL/ services/api/v1/documents/ unassign/system
	POST	Remove sparepart	Application_URL/ services/api/v1/documents/ unassign/sparepart
	POST	Remove equipment	Application_URL/ services/api/v1/documents/ unassign/equipment
Document Assign	POST	Assign a model	Application_URL/ services/api/v1/documents/ assign/model
	POST	Assign an announcement	Application_URL/ services/api/v1/documents/ assign/announcement
	POST	Assign step	Application_URL/ services/api/v1/documents/ assign/steps
	POST	Assign a location	Application_URL/ services/api/v1/documents/ assign/functionallocation
	POST	Assign a functional location	Application_URL/ services/api/v1/documents/ assign/floc
	POST	Assign improvement case	Application_URL/ services/api/v1/documents/ assign/improvementcase
	POST	Assign model request	Application_URL/ services/api/v1/documents/ assign/modelrequest
	POST	Assign a system	Application_URL/ services/api/v1/documents/ assign/system
	POST	Assign spareParts	Application_URL/ services/api/v1/documents/ assign/spareParts
	POST	Assign equipment	Application_URL/ services/api/v1/documents/ assign/equipment

7.1 Additional Information

The following table provides information about the fields of the request payload, along with the corresponding data types, and the data type descriptions:

Field Name	Data Type	Description and Possible Values
documentId (Mandatory)	String	Specifies the document ID the system generates for a document.
phaseCode (Mandatory)	Integer	Specifies the phaseCode for multiple phases. For more information on possible values, see Table Phase Codes and Category Codes.
categoryCode (Mandatory)	Integer	Specifies categorycode that corresponds to a phasecode. For more information on possible values, see Table Phase Codes and Category Codes.
Description (Optional)	String	Specifies the description of the document.
fileName	String	Specifies the name of the file with extension.

The following table provides information about the fields of the response payload, the corresponding data types, and the data type descriptions. To avoid repetition, fields that are also included in the request payload are excluded from the table.

Field Name	Data Type	Description and Possible Values
title, name	String	Specifies the name or a title of the document.
Phase	String	Specifies the phase in the lifecycle of a model or an equipment that the document is relevant for. For more information on possible values, see Table Phase Codes and Category Codes
Category	String	Specifies the category that corresponds to a phase in the lifecycle of a model or an equipment that the document is relevant for. For more information on possible values, see the Phase Codes and Category Codes table.

Field Name	Data Type	Description and Possible Values
fileSize	Integer	Specifies the size of a document.
fileSizeUnit	Integer	Specifies the unit used to measure the file size.
mimeType	String	Specifies the mimetype of the document.

Category Codes

Category Code	Category
1	Identification
2	Technical Specification
3	Drawings, Plans
4	Components
5	Certificates
6	Assembly, Disassembly
7	Operation
8	General safety
9	Inspection, Maintenance
10	Repair
11	Contract
12	Others
13	Firmware
14	Spare parts
15	Decommissioning Plan
16	Decommissioning Checklist
17	Decommissioning Report
18	Dismantling Guide

Phase Codes

Phase Code	Phase
1	Design
2	Acquisition
3	Installation
4	Operation
5	Maintenance
6	Decommissioning
7	Dismantle
8	Disposal

Supported MIME Types for Documents

File Extension	MIME Type
.txt	text/plain
.jpg	image/jpeg
.jpeg	image/jpeg
.jpe	image/jpeg
.png	image/png
.bmp	image/jpeg
.dib	image/jpeg
.bmp	image/x-windows-bmp
.tif	image/jpeg
.tiff	image/jpeg
.jfif	image/jpeg
.gif	image/gif
.doc	application/msword
.docx	application/vnd.openxmlformats-officedocument.wordprocessingml.document
.docm	application/vnd.ms-word.document.macroenabled.12

File Extension	MIME Type
.dotx	application/vnd.openxmlformats-officedocument.wordprocessingml.template
.rtf	application/rtf
.pdf	application/pdf
.pptx	application/vnd.openxmlformats-officedocument.presentationml.presentation
.pptm	application/vnd.ms-powerpoint.presentation.macroEnabled.12
.ppt	application/vnd.ms-powerpoint
.odp	application/vnd.oasis.opendocument.presentation
.ods	application/vnd.oasis.opendocument.spreadsheet
.odt	application/vnd.oasis.opendocument.text
.xps	application/vnd.ms-xpsdocument
.vds	application/octet-stream
.dif	video/x-dv
.prn	application/octet-stream
.dib	application/octet-stream
.csv	text/csv
.slk	application/sylk
.xlsb	application/vnd.ms-excel.sheet.binary.macroenabled.12
.xlsm	application/vnd.ms-excel.sheet.macroenabled.12
.xltx	application/vnd.openxmlformats-officedocument.spreadsheetml.template
.xls	application/vnd.ms-excel
.xltn	application/vnd.ms-excel.template.macroEnabled.12
.xlsx	application/vnd.openxmlformats-officedocument.spreadsheetml.sheet
.html	application/html
.xsl	application/xml

File Extension	MIME Type
.xdp	application/vnd.adobe.xdp+xml
.htm	application/html
.avi	video/avi
.mp4	video/mp4
.mov	video/quicktime
.wmv	video/x-ms-asf
.wav	audio/wav
.mp3	audio/mpeg3
.aml	application/AML
.url	application/octet-stream
.xml	application/xml
.dwg	application/acad
.dxf	application/dxf
.sat	application/sat
.stp	application/step
.bag	application/octet-stream
.h4	application/x-hdf
.h5	application/x-hdf5
.he5	application/octet-stream
.hdf4	application/x-hdf
.hdf5	application/x-hdf5
.hdf	application/x-hdf
.msz	application/octet-stream
.fbx	application/octet-stream
ELINK	application/xml
.out	application / x-out
.mon	application/octet-stream
.zip	application/zip
.tar	application/x-tar
.tgz	application/x-gzip
.deb	application/vnd.debian.binary-package

File Extension	MIME Type
.aasx	application/octet-stream
.amlx	application/octet-stream
.catpart	application/octet-stream
.catproduct	application/octet-stream
.diag	application/octet-stream
.gate	application/octet-stream
.max2	application/octet-stream
.mli	application/octet-stream
.msg	application/octet-stream
.pad	application/octet-stream
.rfrec	application/octet-stream
.rfreccfg	application/octet-stream
.simcmp	application/octet-stream
.simmac	application/octet-stream
.simtpl	application/octet-stream
.spp	application/octet-stream
.uap	application/octet-stream
.ap14	application/octet-stream
.ap16	application/octet-stream
.bin	application/octet-stream
ctconf	Data Exchange File
bmp	image/x-ms-bmp
mdf	application/octet-stream
.ldf	application/octet-stream
.bak	Data Exchange File
.nvrnm	application/octet-stream
.vbox	application/octet-stream
.vmsd	application/octet-stream
.vmx	application/octet-stream
vmxf	application/octet-stream

8 Equipment

An item of an equipment is a physical instance of a model.

In SAP Asset Intelligence Network, an item of equipment comprises of the following information:

- **Attributes:** Properties of the item of equipment.
- **Components:** Detachable objects of an item of equipment.
- **Assignments:** Objects assigned to the item of equipment such as instructions, documents, to name a few.
Header: Primary information which describes an item of equipment such as name, description, installation information, to name a few.
Public APIs are available for each of these specific sections of the item of equipment.

An item of equipment can be in the following states:

- Unpublished
- In Revision
- Published

i Note

To edit an item of equipment, you must move it to the In Revision state.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles EQUIPMENT_READ, EQUIPMENT_EDIT, and EQUIPMENT_DELETE assigned to your user ID.
- **API hub link:** [Equipment API](#) 

Methods

Group	Method	Actions	URI
List and Header	POST	Creates an equipment	Application_URL/ services/api/v1/equipment
<p>i Note</p> <p>By default, the external System relevance flag (<isExternalSystemRelevant>) is set to True. In this case, you must specify the External System (<externalSystemID>) if you have maintained more than one external ERP system.</p> <p>If you have defined external ERP systems and do not want to sync data with external ERP systems, then you must set the external System relevance flag (<isExternalSystemRelevant>) to False.</p>			
	GET	Lists equipments based on filter query parameter	Application_URL/ services/api/v1/equipment
	GET	Returns the count of equipments based on filter query parameter	Application_URL/ services/api/v1/equipment/ \$count
	GET	Retrieves equipment header information for the specific equipment	Application_URL/ services/api/v1/equipment({equipmentId})/ header
	PUT	Update header details of the specific equipment	Application_URL/ services/api/v1/equipment({equipmentId})/ header
	DELETE	Deletes the specific equipment	Application_URL/ services/api/v1/equipment({equipmentID})
	PUT	Publish one or more equipments	Application_URL/ services/api/v1/equipment/ publish

Group	Method	Actions	URI
	PUT	Create new revision of the specified equipment	Application_URL/ services/api/v1/equipment/{equipmentId}/revise
	PUT	Publish the specified equipment	Application_URL/ services/api/v1/equipment/{equipmentId}/publish
	PUT	Update the equipment lifecycle	Application_URL/ services/api/v1/equipment/{equipmentId}/changelifecycle
	GET	Retrieves the list of templates associated with the equipment	Application_URL/ services/api/v1/equipment/{equipmentId}/templates
	PUT	Modify the templates associated with the equipment	Application_URL/ services/api/v1/equipment/{equipmentId}/templates
	PUT	Transfer equipment to other organization	Application_URL/ services/api/v1/equipment/{equipmentID}/transfer
	GET	Retrieves equipment header information for the specific equipment.	Application_URL/ services/api/v1/equipment/internalid/{internalid}/equipment_header
		<div style="border: 1px solid #ccc; background-color: #f9f9f9; padding: 10px; margin: 10px 0;"> <p>i Note</p> <p>This API will be deprecated soon. Use /equipment/internalid/{name}/header instead.</p> </div>	
	GET	Retrieves list of equipment headers for the specific equipment name	Application_URL/ services/api/v1/equipment/internalid/{name}/header
Attribute Values	GET	Retrieves attribute values of the equipment	Application_URL/ services/api/v1/equipment/{equipmentId}/values
	PUT	Modify attribute values of an equipment	Application_URL/ services/api/v1/equipment/{equipmentId}/values
	PATCH	Updates attribute values of a equipment	Application_URL/ services/api/v1/equipment/{equipmentId}/values

Group	Method	Actions	URI
	GET	Retrieves attribute values of the equipment	Application_URL/ services/api/v1/base- line({baselineId})/equip- ment({equipmentId})/values
Component Structure	GET	Retrieves component struc- ture of an equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/ components
	PUT	Updates component struc- ture of the equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/ components
	PATCH	Updates component struc- ture of the equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/ components
	GET	Retrieves immediate func- tional location parent to an equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/floc- Parent
Equipment Object Associa- tions	GET	Retrieves the documents as- signed to an equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/docu- ments
	POST	Assign documents to an equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/docu- ments
	GET	Retrieves all instructions as- signed to the equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/ instructions
	GET	Retrieves all the spareparts assigned to the equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/spare- parts
	GET	Retrieves all the workorders assigned to the equipment	Application_URL/ services/api/v1/equip- ment({equipmentId})/work- orders
	PUT	Add or remove multiple in- structions to or from multiple equipment	Application_URL/ services/api/v1/equipment/ instructions
	PUT	Add/remove/update the quantities of multiple parts assigned to a piece of equip- ment	Application_URL/ services/api/v1/equipment/ spareparts

Group	Method	Actions	URI
Other	GET	Retrieve all timeline events of the equipment	Application_URL/ services/api/v1/equipment({equipmentId})/time-line
	POST	Create future events associated with the equipment	Application_URL/ services/api/v1/equipment({equipmentID})/futureEvent
	GET	Get all organizations in the system	Application_URL/ services/api/v1/organizations
	GET	Get organizations with specific role	Application_URL/ services/api/v1/organizations/byrole
	GET	Get the count of all organizations	Application_URL/ services/api/v1/organizations/\$count
	GET	Get the enumeration values	Application_URL/ services/api/v1/enumeration
	GET	Get the enumeration currency codes	Application_URL/ services/api/v1/enumeration/currencycodes
	GET	Get a particular value of an enumeration type	Application_URL/ services/api/v1/enumerationvalue
	GET	Search an equipment based on its attribute values	Application_URL/ services/api/v1/equipment/attributeSearch
	PUT	Update installation location of an equipment.	Application_URL/ services/api/v1/equipment({equipmentId})/reset/installationLocation
	GET	Read installation location details of functional location object.	Application_URL/ services/api/v1/equipment({equipmentId})/reset/installationLocation
	PUT	Update installation location details of an equipment.	Application_URL/ services/api/v1/equipment({equipmentId})/installationLocation
	GET	Read installation location details of functional location object.	Application_URL/ services/api/v1/equipment({equipmentId})/installationLocation

Group	Method	Actions	URI
	GET	Fetching equipment based in Internal ID	Application_URL/ services/api/v1/equipment/ internalid/{internalid}/exter- nal_info
	PUT	Equipment's IoT sync config- uration Update	Application_URL/ services/api/v1/ equipment/iot/sync/config
	GET	Get Business Partner Ids	Application_URL/ services/api/v1/equipment /uid
Indicator Values	POST	Configure indicator with thresholds and other proper- ties of equipment	Application_URL/ services/api/v1/equip- ment({id})/indicatorconfigu- ration
	GET	Get values for particular indi- cator passed as query pa- rameter	Application_URL/ services/api/v1/indicator- configuration
	PUT	Edit the configuration of spe- cific indicator	Application_URL/ services/api/v1/indicator- configuration({id})
	POST	Specify the values of an indi- cator	Application_URL/ services/api/v1/indicatorval- ues
	GET	Get values for a particular in- dicator	Application_URL/ services/api/v1/indicatorval- ues
	PUT	Update a new value to an in- dicator	Application_URL/ services/api/v1/indicator- value({id})
	DELETE	Delete the indicator value	Application_URL/ services/api/v1/indicator- value({id})
	GET	Read indicator values for an equipment	Application_URL/ services/api/v1/equip- ment({id})/indicatorvalues
	PUT	Reset the value of indicators associated with an equip- ment	Application_URL/ services/api/v1/equip- ment({equipmentId})/reset/ indicatorvalues
Nested Structure Indicator	GET	Get nested structure indica- tor metadata.	Application_URL/ services/api/v1/equip- ment({equipmentId})/ nested/structure/indicator
	POST	Post nested structure indica- tor values.	Application_URL/ services/api/v1/nested/ structure/indicatorvalues

Group	Method	Actions	URI
	POST	Read nested structure indicator values.	Application_URL/ services/api/v1/nested/ structure/read/indicatorval- ues
Equipment Search	PUT	Search for available equipment based on search and/or filter conditions.	Application_URL/ services/api/v1/search/ equipment
Baseline	POST	Creates a baseline	Application_URL/ services/api/v1/equip- ment({equipmentId})/base- lines
	GET	Get all baselines for an equip- ment	Application_URL/ services/api/v1/equip- ment({equipmentId})/base- lines
	PUT	Edit baseline description	Application_URL/ services/api/v1/equip- ment({equipmentId})/base- line({baselineId})
	GET	Get all baselines for an equip- ment	Application_URL/ services/api/v1/base- line({baselineId})
	GET	Retrieves equipment header information for the specific baseline and equipment	Application_URL/ services/api/v1/base- line({baselineId})/equip- ment({equipmentId})/ header
	GET	Retrieves component struc- ture of an equipment	Application_URL/ services/api/v1/base- line({baselineId})/equip- ment({equipmentId})/ components
	GET	Retrieves the list of tem- plates associated with the equipment	Application_URL/ services/api/v1/base- line({baselineId})/equip- ment({equipmentId})/ templates
	GET	Retrieves all the spareparts assigned to the equipment	Application_URL/ services/api/v1/base- line({baselineId})/equip- ment({equipmentId})/ GetAssignedSparePartsForA- nEquipment

Group	Method	Actions	URI
	GET	Retrieves the documents assigned to an equipment	Application_URL/services/api/v1/baseline({baselineId})/equipment({equipmentId})/GetAssignedDocumentsOfEquipment
	GET	Get all baseline operations for an equipment	Application_URL/services/api/v1/equipment({objectId})/asyncoperations
	GET	Get baseline operation for specified equipment id and operation id	Application_URL/services/api/v1/equipment({objectId})/asyncoperation({operationId})

8.1 Equipment Search

Constructing Search Request

Free-Text Search

You can search for equipment based on the following equipment fields: **internalId**, **longDescription**, **shortDescription**, **batchNumber**, **serialNumber**, **secondaryKey**, **tin**, **procurementNumber**, **manufacturerPartNumber**, **uid**.

Example: If a piece of equipment has an internal ID: ABC_EQU, the search-text **ABC_**, **A**, **AB**, **ABC_E**, and so on, will return the equipment details of **ABC_EQU** in the response.

Sample Request:

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": "ABC_",
  "orderBy": {
    "path": "changedOn",
    "descending": true
  },
  "select": [],
  "attributes": []
}
```

Sorting Search Results

The search results can be sorted by defining the 'orderBy' condition. Supported equipment fields for orderBy are: **internalId, changedOn, createdOn, publishedOn, locationName, className, parentName, subclassName, equipmentId**.

i Note

Default Value for 'orderBy' will be sorting based on **changedOn** in the descending order.

orderBy definition takes structure:

Sample Code

```
{
  "path": "<equipment_header_field>",
  "descending": <true/false>
}
```

Constructing Filters for Search

- **Basic Filter Structure:**

Filters are used to define the search conditions. A basic filter can be defined as:

Sample Code

```
{
  "operator": "<operator>",
  "value1": "<value>",
  "value2": "<value>", # Required when defining a range
  "path": "<field>"
}
```

Filter conditions support the following:

- Operators
 - "EQ" - Equals
 - "GT" - Greater than
 - "GE" - Greater than or Equal to
 - "LT" - Less than
 - "LE" - Less than or Equal to
 - "IN" - Present in a list/array
 - "BT" - Between a range (range is defined by setting value1 & value2 fields)
- Fields: - **equipmentId, internalId, templates, classId, subclassId, lifecycle, uid, operatorId, clientId, manufacturerId, modelId, locationId, status, productRelevance, externalSystemId, planningPlant, maintenancePlant, plannerGroup, maintenanceWorkCenter, plant, changedOn, additionalBusinessPartners, value1, value2, attributeId, attributes, manufacturerName.**

i Note

The filter value defined in "value1"/"value2" is not free text enabled. The exact term must be entered for the search to return the desired results.

Examples:

1. Filter for an attribute.

Sample Code

```
{
  "operator": "EQ",
  "value1": "D591D46E4C26451B9FC39765954F492C",
  "path": "attributeId"
}
```

2. Filter all equipment published between two dates.

Sample Code

```
{
  "operator": "BT",
  "value1": 1592213215896,
  "value2": 1608024415896,
  "path": "publishedOn"
}
```

3. Filter all equipment having values v1 or v2 using In operator.

Sample Code

```
{
  "path": "attributeId",
  "value1": "attributeId",
  "operator": "EQ"
},
{
  "path": "value1",
  "values": ["v1", "v2"],
  "operator": "IN"
}
```

- **Combining Multiple Filters:**

Multiple filters can be combined to form multiple or nested filters.

- Multiple filters are formed when more than one filter is defined in a level connected by a logical operation (Logical AND or Logical OR).
- Nested filters are formed when a hierarchy of filters is defined at multiple levels connected by a logical operation (Logical AND or Logical OR).

The basic structure is as follows:

Sample Code

```
{
  "filters": [{
    "operator": "<operator>",
    "value1": "<value>",
    "path": "<path>"
  },
  {
    "operator": "<operator>",
    "value1": "<value>",
    "path": "< path>"
  }
],
```

```
"and": true. # Logical AND/OR
}
```

Example:

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": null,
  "orderBy": {
    "path": "changedOn",
    "descending": true
  },
  "filter": {
    "filters": [
      {
        "filters": [
          {
            "operator": "EQ",
            "value1": "0E261150A7AA4934BE493D020E54C18C",
            "path": "attributeId"
          },
          {
            "operator": "EQ",
            "value1": "true",
            "path": "value"
          }
        ],
        "and": true
      },
      {
        "filters": [
          {
            "operator": "BT",
            "value1": 1606258801000,
            "value2": 1606345199000,
            "path": "changedOn"
          }
        ]
      }
    ],
    "and": false
  }
}
```

Pagination

Pagination can be achieved by simultaneously setting the top and skip request options with every request.

Note

The default value of top and skip will be 20 and 0.

The supported maximum value of top + skip is 1000.

Pagination using SearchAfter

For pagination in AWS beyond 1000 objects, use the property **searchAfter**. This value will be the part of every response and must be sent in the subsequent call to get the next batch of records.

i Note

- The **orderBy** field won't be applied in case **searchAfter** is being used.
- The skip parameter won't be applied if **searchAfter** is being used.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": "ABC_",
  "orderBy": null,
  "select": [],
  "attributes": [],
  "searchAfter": ""
}
```

Example:

Sample Code

```
{
  "count": 0,
  "value": [],
  "searchAfter": "4BC02F4450CE4321B3112937346944C2"
}
```

Response Fields Selection

The 'select' parameter in the request body determines the fields to be present in the response.

The supported fields are : **equipmentId**, **clientId**, **status**, **attributes**, **internalId**, **image**, **description**, **uid**, **subclassId**, **subclassName**, **manufacturerId**, **manufacturerName**, **templates**, **modelId**, **modelName**, **lifecycle**, **operatorId**, **additionalBusinessPartners**, **changedOn**, **createdOn**, **publishedOn**, **classId**, **className**, **completeness**, **locationId**, **locationName**, **parentId**, **parentName**, **batchNumber**, **serialNumber**, **tin**, **procurementNumber**, **manufacturerPartNumber**, **secondaryKey**, **productRelevance**, **planningPlant**, **maintenancePlant**, **plannerGroup**, **maintenanceWorkCenter**, **plant**, **externalSystemId**, **lifeCycle**, **operator**, **source**, **buildDate**.

If 'select' is not defined in the request, the response will contain only the fields **equipmentId** and **internalId**.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "select": ["equipmentId", "internalId", "status", "classId", "subclassName"],
  "attributes": []
}
```

Filtering Attributes

The 'attributes' parameter in the request body defines the attributes to be considered for the search operation. It takes a list of valid attribute IDs for which the user has authorization to access. Any attributes, which are not mentioned under the 'attributes' section will not be part of the response.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "select": ["equipmentId",
    "internalId", "status", "classId", "subclassName"],
  "attributes": [
    "21858B573E3A49A9AA93F82521B0D13D", "32969C64F4B49A9AA93F82521B0D13D"]
}
```

Example Scenarios

1. Searching with attribute values

Scenario: Search for all equipment having an attribute with ID '21858B573E3A49A9AA93F82521B0D13D' and value as '23'.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": null,
  "orderBy": {
    "path": "changedOn",
    "descending": true,
  },
  "attributes": [
    "21858B573E3A49A9AA93F82521B0D13D"
  ],
  "filter": {
    "filters": [
      {
        "filters": [
          {
            "operator": "EQ",
            "value1": "21858B573E3A49A9AA93F82521B0D13D",
            "path": "attributeId"
          },
          {
            "operator": "EQ",
            "value1": "23",
            "path": "value1"
          }
        ],
        "and": true
      }
    ],
    "and": true
  }
}
```

2. Free-text search combined with multiple filters

Scenario: Search for all equipment with an internal ID that starts with "abc", and the last changed date between two dates and attribute ID as '9DC51C442950460EBOAD947A7A6DD05E'.

Sample Code

```
{
```

```

"top": 20,
"skip": 0,
"search": "abc",
"orderBy": {
  "path": "changedOn",
  "descending": true,
},
"attributes": [ "9DC51C442950460EB0AD947A7A6DD05E" ],
"filter": {
  "filters": [
    {
      "filters": [
        {
          "operator": "EQ",
          "value1": "0E261150A7AA4934BE493D020E54C18C",
          "path": "attributeId"
        }
      ]
    },
    {
      "filters": [
        {
          "operator": "BT",
          "value1": 1606258801000,
          "value2": 1606345199000,
          "path": "changedOn"
        }
      ]
    }
  ],
  "and": false
}
}

```

Limitations

- This application is only available on SAP BTP, Cloud Foundry environment (Amazon Web Services (AWS) and Azure).
- Sorting by attribute values is not supported in the Azure cloud.
- Search on attribute values from location, which are assigned to the equipment are not supported. You can search equipment based on the model attributes. You can also search equipment based on attribute codelist values.
- Filter on attributes of type picture, enum, attributes with codelists of type date, numeric, and numeric flexible having multiselect option and along with unit of measure are not supported.
- Multilanguage string attribute won't be displayed as per your logon language.
- Search on attributes with a unit of measure is only supported with the actual value maintained. The automatic value conversion to your preferred metric even when the value is maintained in a different metric is not supported. For example, if you have saved 10 Kilograms as a value and try to search for 22 Pounds, you won't get the results, even though the value is the same in different metrics.
- The search results are 'near real-time', there might be a slight delay between when the changes are made and when they are available in the search results.
- Partial text search on string attributes is not supported.
- Azure Limitations:

- OrderBy: 'parentName' is not supported.
- Filters & Select: planningPlant, maintenancePlant, plannerGroup, maintenanceWorkCenter, plant, parentId, parentName are not supported.

8.2 Creating a Baseline

You use this API to create a baseline for a piece of equipment.

Request

- *HTTP Method*: POST
- *URL*: Application_URL/services/api/v1/equipment({equipmentid})/baselines
- *Authorization*: You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID in the SAP Cloud Platform account.

Operation	Parameters
Create Baseline	<ul style="list-style-type: none"> • equipmentid: string, Equipment ID • baselineDescription <ul style="list-style-type: none"> • type: string, maxLength=256, Textual description of the baseline to be created. • context: string, maxLength: 40 The asset for which the baseline is created. Equipment context will be assigned by default if it is not specified.

Response

If you have appropriate authorizations to create a baseline, a baseline is created. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#equipment__equipmentid__baselines_post file.

8.3 View all the Baselines Created for an Equipment

You use this API to retrieve all the baselines created for an equipment.

Request

- *HTTP Method:* GET
- *URL:* `Application_URL/services/api/v1/equipment({equipmentid})/baselines`
- *Authorization:* You must have the roles `BASELINE_READ` or `BASELINE_EDIT` assigned to your user ID in the SAP Cloud Platform account.

Response

If you have appropriate authorizations to view baselines, a list of baselines created for the equipment is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#equipment__equipmentid__baselines_get file.

8.4 Editing a Baseline Description

You use this API to edit Baseline description for equipment.

Request

- *HTTP Method:* PUT
- *URL:* `Application_URL/services/api/v1/equipment({equipmentid})/baseline({baselineid})`
- *Authorization:* You must have the roles `BASELINE_READ` or `BASELINE_EDIT` assigned to your user ID in the SAP Cloud Platform account.

Operation	Parameters
Edit Baseline	<ul style="list-style-type: none"> • equipmentid: string, Equipment ID • baselineid: string, baselineID • baselineDescription <ul style="list-style-type: none"> • type: string, maxlength=256, Textual description of the baseline to be created.

Response

If you have appropriate authorizations to edit baselines, you can edit and update the baseline description of the equipment. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#equipment__equipmentid__baseline__baselineid__put file.

8.5 View a Specific Baseline

You use this API to retrieve the specific baseline created for an equipment.

Request

- *HTTP Method:* GET
- *URL:* Application_URL/services/api/v1/baseline({baselineid})
- *Authorization:* You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID in the SAP Cloud Platform account.

Response

If you have appropriate authorizations to view baselines, the baseline with the specified ID is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#baseline__baselineid__get file.

8.6 Retrieve Header Information for a Baselined Equipment

You use this API to retrieve header information for a specific equipment which has been baselined.

Request

- *HTTP Method:* GET
- *URL:* Application_URL/services/api/v1/baseline({baselineid})/equipment({equipmentid})/header
- *Authorization:* You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID in the SAP Cloud Platform account.

Response

If you have appropriate authorizations to view baselines and equipment, the header information of the equipment for which the specified baseline was created is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#baseline__baselineid__equipment__equipmentid__header_get file.

8.7 View Components Information

You use this API to retrieve components information for a specific equipment which has been baselined.

Request

- *HTTP Method:* GET
- *URL:* Application_URL/services/api/v1/baseline({baselineid})/equipment({equipmentid})/components
- *Authorization:* You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID in the SAP Cloud Platform account.

Response

If you have appropriate authorizations to view baselines and equipment, the components information of the equipment for which the specified baseline was created is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#baseline__baselineid__equipment__equipmentid__components_get file.

8.8 View Template Information

You use this API to retrieve templates information for a specific equipment which has been baselined.

Request

- *HTTP Method:* GET
- *URL:* Application_URL/services/api/v1/baseline({baselineid})/equipment({equipmentid})/templates
- *Authorization:* You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID in the SAP Cloud Platform account.

Response

If you have appropriate authorizations to view baselines and equipment, the templates information of the equipment for which the specified baseline was created is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#baseline__baselineid__equipment__equipmentid__templates_get file.

8.9 View Values Information

You use this API to retrieve values information for a specific equipment which has been baselined.

Request

- *HTTP Method:* GET
- *URL:* Application_URL/services/api/v1/baseline({baselineid})/equipment({equipmentid})/values
- *Authorization:* You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID in the SAP Cloud Platform account.

Response

If you have appropriate authorizations to view baselines and equipment, the values information of the equipment for which the specified baseline was created is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#baseline__baselineid__equipment__equipmentid__values_get file.

8.10 View Spare Parts Information

You use this API to retrieve spare parts information for a specific equipment which has been baselined.

Request

- *HTTP Method:* GET
- *URL:* Application_URL/services/api/v1/baseline({baselineid})/equipment({equipmentid})/GetAssignedSparePartsForAnEquipment
- *Authorization:* You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID in the SAP Cloud Platform account.

Response

If you have appropriate authorizations to view baselines and equipment, the spare parts information of the equipment for which the specified baseline was created is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#baseline__baselineid__equipment__equipmentid__GetAssignedSparePartsForAnEquipment_get file.

8.11 View Documents Information

You use this API to retrieve documents information for a specific equipment which has been baselined.

Request

- *HTTP Method:* GET
- *URL:* Application_URL/services/api/v1/baseline({baselineid})/GetAssignedDocumentsOfEquipment
- *Authorization:* You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID in the SAP Cloud Platform account.

Response

If you have appropriate authorizations to view baselines, the documents information of the equipment for which the specified baseline was created is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#baseline__baselineid__equipment__equipmentid__GetAssignedDocumentsOfEquipment_get file.

8.12 View a Specific Async Operation

You use this API to read about a specific async operation associated for an equipment baseline creation.

Request

- *HTTP Method:* GET
- *URL:* Application_URL/services/api/v1/equipment({objectId})/asyncooperation({operationId})
- *Authorization:* in the SAP Cloud Platform account You must have the roles BASELINE_READ or BASELINE_EDIT assigned to your user ID.

Response

You must have the roles `BASELINE_READ` or if you have appropriate authorizations to view baselines, the specific operation associated for an equipment baseline creation is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#equipment__objectId__asyncoperations_get file.

8.13 View all Async Operations

You use this API to read all async operations associated for an equipment baseline creation. Operation in baseline context can be defined as the active process during which baseline creation is performed. The baseline's creation depends on the success of the operation.

Request

- *HTTP Method:* GET
- `Application_URL/services/api/v1/equipment({objectId})/asyncoperations`
- *Authorization:* You must have the roles `BASELINE_READ` or `BASELINE_EDIT` assigned to your user IDURL: in the SAP Cloud Platform account.

Response


If you have appropriate authorizations to view baselines, all async operations associated for an equipment baseline creation is displayed. Otherwise an error message is displayed.

For more information, see https://help.sap.com/doc/3eb9f8a3065d419ea1e6e5fcb8fd773d/2011/en-US/baseline.html#equipment__objectId__asyncoperation__operationId__get file.

9 Equipment Requests

This request triggers a workflow that brings together the relevant stakeholders to collaborate and help you to find out the relevant equipment recommendation for possible combination of ModelID, Serial Number, Technical ID Number, Operator Equipment ID, Procurement Number, UID.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles MODEL_REQUEST_READ, MODEL_REQUEST_EDIT, or MODEL_REQUEST_DELETE assigned to your user ID.
- **API hub link:** [Equipment Request API](#) 

Methods

Method	Actions	URI
POST	Creates a equipment request	Application_URL/services/api/v1/equipmentrequests
GET	Lists equipment requests based on filter query parameter	Application_URL/services/api/v1/equipmentrequests
GET	Returns the count of equipment requests based on filter query parameter	Application_URL/services/api/v1/equipmentrequests/\$count
GET	Lists all the configured business partners as providers	Application_URL/services/api/v1/equipmentrequests/providers
PUT	Update status of the specific equipment Request	Application_URL/services/api/v1/equipmentrequests/{equipmentRequestid}/action/{actionId}
PUT	Update details of the specific equipment request	Application_URL/services/api/v1/equipmentrequests/{equipmentRequestid}
GET	Gets basic details of a equipment request	Application_URL/services/api/v1/equipmentrequests/{equipmentRequestid}
DELETE	Deletes the specific equipment request	Application_URL/services/api/v1/equipmentrequests/{equipmentRequestid}
GET	Lists line items that are assigned to a particular equipment request	Application_URL/services/api/v1/equipmentrequests/{equipmentRequestid}/items
GET	Lists documents that are assigned to a particular equipment request	Application_URL/services/api/v1/equipmentrequests/{equipmentRequestid}/documents

Method	Actions	URI
POST	Post a comment on the equipment request	Application_URL/services/api/v1/equipmentrequests/{equipmentRequestId}/comments
DELETE	Deletes the comment for a given equipment request	Application_URL/services/api/v1/equipmentrequests/{equipmentRequestId}/comments/{commentId}

10 External IDs

External ID is the unique object ID of an external system that you want to map with the object ID in SAP Asset Intelligence Network. The external ID is the means to map objects in SAP Asset Intelligence Network with an external system.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles COMPANYPROFILE_READ, COMPANYPROFILE_DELETE, or COMPANYPROFILE_EDIT assigned to your user ID.
- **API hub link:** [External ID API](#)

Methods

Group	Method	Actions	URI
External Systems	GET	Retrieves all the external systems	Application_URL/services/api/v1/external/systems
	POST	Creates an external system	Application_URL/services/api/v1/external/systems
	PUT	Updates an external system	Application_URL/services/api/v1/external/systems
External Object Types	GET	Retrieves all the external object types	Application_URL/services/api/v1/externalobjecttypes
	PUT	Updates the External Object type	Application_URL/services/api/v1/externalobjecttypes
	POST	Creates external object type	Application_URL/services/api/v1/externalobjecttypes
	DELETE	Deletes the external object type	Application_URL/services/api/v1/externalobjecttypes
External Ids	PUT	Update the external ID	Application_URL/services/api/v1/object/externalid
	PUT	Delete the external ID	Application_URL/services/api/v1/object/externalid/Delete

Group	Method	Actions	URI
	POST	Maintain external IDs in one or more external systems	Application_URL/ services/api/v1/object/ externalids
	GET	Retrieves all the associated external ID	Application_URL/ services/api/v1/objectid/ externalid({objectId})
	GET	Retrieves all the associated internal objects for specified external ID	Application_URL/ services/api/v1/objectsid/ ainobjects({externalId})


11 Equipment Sync

You can use this feature to check the current or intermediate status of all equipment that is in the process of being synchronized with SAP Internet of Things.

By default, the logs display the synchronization status of a piece of equipment for the last one hour. The various synchronization statuses are as below:

- **SYNC IN PROGRESS:** This status appears when the events are being processed.
- **SYNC FAILURE:** This status appears for a failed synchronization.
- **SYNC COMPLETED:** This status appears after the synchronization completes successfully.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles EQUIPMENT_READ assigned to your user ID.
- **API hub link:** [Equipment Sync API](#) 

Methods


Method	Actions	URI
GET	Lists equipment that are synced to SAP Internet of Things and its sync status.	Application_URL/services/api/v1/sync/log

12 Failure Modes

A failure mode is the manner in which the item or operation potentially fails to meet or deliver the intended function and associated requirements.

This API allows you to create, update, retrieve, and delete effects and causes. Cause is the specific reason for the failure. An effect is the consequence of the failure on the system or end user.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles FAILURE_MODE_READ, FAILURE_MODE_EDIT, or FAILURE_MODE_DELETE assigned to your user ID.
- **API hub link:** [Failure Modes API](#) 

Methods

Group	Method	Actions	URI
Failure Modes	GET	Returns details of failure modes	Application_URL/services/api/v1/failuremodes
	POST	Creates a failure mode	Application_URL/services/api/v1/failuremodes
	PUT	Update failure mode details	Application_URL/services/api/v1/failuremodes
	DELETE	Deletes a failure mode	Application_URL/services/api/v1/failuremodes/delete
	PUT	Publishes a failure mode	Application_URL/services/api/v1/failuremodes/publish
	GET	Returns details of a failure mode	Application_URL/services/api/v1/failuremodes/{failureModelId}
	PUT	Assigns object to failure mode	Application_URL/services/api/v1/failuremodes/{failureModelId}/assign
	PUT	Creates new cause	Application_URL/services/api/v1/failuremodes/{failureModelId}/causes

Group	Method	Actions	URI
	POST	Copy a failure mode	Application_URL/ services/api/v1/failuremodes/{failureModelId}/copy
	GET	Return details of detection methods	Application_URL/ services/api/v1/failuremodes/{failureModelId}/detections
	PUT	Save detection methodsSave detection methods	Application_URL/ services/api/v1/failuremodes/{failureModelId}/detections
	PUT	Dissociates object from failure mode	Application_URL/ services/api/v1/failuremodes/{failureModelId}/dissociate
	GET	Returns details of failure mode	Application_URL/ services/api/v1/failuremodes/{failureModelId}/figures
	PUT	Saves failure mode	Application_URL/ services/api/v1/failuremodes/{failureModelId}/figures
	GET	Returns details of failure mode assigned objects	Application_URL/ services/api/v1/failuremodes/{failureModelId}/objects
	PUT	Creates new revision of failure mode	Application_URL/ services/api/v1/failuremodes/{failureModelId}/revise
	PUT	Assigns failure mode to object	Application_URL/ services/api/v1/objects/{objectID}/assign
	PUT	Dissociates failure mode from object	Application_URL/ services/api/v1/objects/{objectID}/dissociate
	GET	Returns details of failure mode for an object	Application_URL/ services/api/v1/objects/{objectType}/{objectID}/failuremodes
Failure Mode Instances	GET	Returns details of failure mode associated with an object	Application_URL/ services/api/v1/failuremodes/{failuremodesID}/{objectType}/{objectID}
	GET	Returns details of spareparts assigned to failure mode associated with an object	Application_URL/ services/api/v1/failuremodes/{failuremodesID}/{objectType}/{objectID}/AssignedSpareparts


Group	Method	Actions	URI
	PUT	Assigns effect to failure mode instance	Application_URL/ services/api/v1/failuremodes/{failuremodesID}/{objectType}/{objectID}/assign
	PUT	Dissociates effect from failure mode instance	Application_URL/ services/api/v1/failuremodes/{failuremodesID}/{objectType}/{objectID}/dissociate
	GET	Returns details and relevance of the failure mode	Application_URL/ services/api/v1/failuremodes/{failuremodesID}/{objectType}/{objectID}/header
	PUT	Updates an inherited failure mode as relevant/not relevant	Application_URL/ services/api/v1/objects/{objectID}/failuremodes/{relevance}
	GET	Returns effect instance descriptions	Application_URL/ services/api/v1/effects/{effectID}/failuremodes/{failuremodesID}/header
	PUT	Updates effect instance descriptions	Application_URL/ services/api/v1/effects/{effectID}/failuremodes/{failuremodesID}/header
Failure Mode Metadata	GET	Returns details of all effects	Application_URL/ services/api/v1/failuremodes/enum/effects
	GET	Returns details of all options	Application_URL/ services/api/v1/failuremodes/enum/options
	GET	Returns details of all patterns	Application_URL/ services/api/v1/failuremodes/enum/patterns
	GET	Returns details of all causes	Application_URL/ services/api/v1/failuremodes/meta/causes
	POST	Creates a cause	Application_URL/ services/api/v1/failuremodes/meta/causes
	DELETE	Delete causes	Application_URL/ services/api/v1/failuremodes/meta/causes
	GET	Returns details of cause	Application_URL/ services/api/v1/failuremodes/meta/causes/{causeID}

Group	Method	Actions	URI
	PUT	Updates cause	Application_URL/ services/api/v1/failuremodes/meta/causes/{causeId}
	PUT	Assigns failuremode to cause	Application_URL/ services/api/v1/failuremodes/meta/causes/{causeId}/map
	GET	Returns failuremode types	Application_URL/ services/api/v1/failuremodes/meta/types
	POST	Creates effect	Application_URL/ services/api/v1/failuremodes/meta/effects
	GET	Returns effect categories	Application_URL/ services/api/v1/failuremodes/meta/effects/categories
	PUT	Delete effects	Application_URL/ services/api/v1/failuremodes/meta/effects/delete
	GET	Returns details of effect	Application_URL/ services/api/v1/failuremodes/meta/effects/effectId
	PUT	Updates effect	Application_URL/ services/api/v1/failuremodes/meta/effects/effectId
Others	GET	Returns enumerations values for paramters as key : [values]	Application_URL/ services/api/v1/failuremodes/meta/enumerations

13 Functions API

You use this feature to define specifications on how the assigned objects are intended to operate.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles FUNCTION_READ, FUNCTION_EDIT, or FUNCTION_DELETE assigned to your user ID.
- **API hub link:** [Functions API](#) 

Methods

Method	Actions	URI
GET	Returns the details of functions	Application_URL/services/api/v1/functions
POST	Creates a function	Application_URL/services/api/v1/functions
DELETE	Delete functions	Application_URL/services/api/v1/functions
GET	Retrieves details of a function	Application_URL/services/api/v1/functions/{functionId}
PUT	Updates function details	Application_URL/services/api/v1/functions/{functionId}
GET	Returns details of function for an object	Application_URL/services/api/v1/objects/{objectId}/functions
PUT	Assigns function to object	Application_URL/services/api/v1/objects/{objectId}/functions/assign
PUT	Unassign function from object	Application_URL/services/api/v1/objects/{objectId}/functions/unassign

14 Functional Locations

A functional location is a physical instance of a model, an object-based structure of a technical system.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles FLOC_READ or FLOC_EDIT or FLOC_DELETE assigned to your user ID.
- **API hub link:** <https://api.sap.com/api/FunctionalLocationAPI/resource>

Methods

Group	Method	Actions	URI
List and Header	POST	Creates a functional location	Application_URL/services/api/v1/floc
	GET	Lists functional locations based on filter query parameter	Application_URL/services/api/v1/floc
	GET	Retrieves functional location header information for the specific functional location	Application_URL/services/api/v1/floc({flocId})/header
	PUT	Update header details of the specific functional location	Application_URL/services/api/v1/floc({flocId})/header
	GET	Retrieves a set of functional locations	Application_URL/services/api/v1/floc/internalid/{internalid}/header
	DELETE	Deletes the specific functional location	Application_URL/services/api/v1/floc({flocID})
	PUT	Publish one or more functional locations	Application_URL/services/api/v1/floc/publish

Group	Method	Actions	URI
	PUT	Create new revision of the specified functional location	Application_URL/ services/api/v1/ floc({flocId})/ revise
	PUT	Publish the specified functional location	Application_URL/ services/api/v1/ floc({flocId})/ publish
	GET	Retrieves the list of templates associated with the functional location	Application_URL/ services/api/v1/ floc({flocId})/ templates
	PUT	Modify the templates associated with the functional location	Application_URL/ services/api/v1/ floc({flocId})/ templates
	PUT	Update the functional location life-cycle	Application_URL/ services/api/v1/ floc({flocId})/ changelifecycle
Attribute Values	GET	Retrieves attribute values of the functional location	Application_URL/ services/api/v1/ floc({flocId})/ values
	PUT	Modify attribute values of a functional location	Application_URL/ services/api/v1/ floc({flocId})/ values
	PATCH	Updates attribute values of a functional location	Application_URL/ services/api/v1/ floc({flocId})/ values
Component Structure	GET	Retrieves component structure of a functional location	Application_URL/ services/api/v1/ floc({flocId})/ components
	PUT	Updates component structure of the functional location	Application_URL/ services/api/v1/ floc({flocId})/ components
	PATCH	Updates component structure of the functional location	Application_URL/ services/api/v1/ floc({flocId})/ components

Group	Method	Actions	URI
Indicator Values	POST	Configure indicator with thresholds and other properties of functional location	Application_URL/ services/api/v1/ floc({id})/indicatorconfiguration
	GET	Get values for particular indicator passed as query parameter	Application_URL/ services/api/v1/ indicatorconfiguration
	PUT	Edit the configuration of specific indicator	Application_URL/ services/api/v1/ indicatorconfiguration({id})
	POST	Specify the values of an indicator	Application_URL/ services/api/v1/ indicatorvalues
	GET	Get values for a particular indicator	Application_URL/ services/api/v1/ indicatorvalues
	PUT	Update a new value to an indicator	Application_URL/ services/api/v1/ indicator-value({id})
	DELETE	Delete the indicator value	Application_URL/ services/api/v1/ indicator-value({id})
	GET	Read indicator values for a functional location	Application_URL/ services/api/v1/ floc({id})/indicatorvalues
	PUT	Reset the value of indicators associated with a functional location	Application_URL/ services/api/v1/ floc({flocId})/ reset/indicatorvalues
Others	PUT	Update installation location of a functional location.	Application_URL/ services/api/v1/ floc({flocId})/ installationlocation
	GET	Read installation location details of functional location object.	Application_URL/ services/api/v1/ floc({flocId})/ installationlocation

Group	Method	Actions	URI
Functional Location Search	PUT	Search for available functional locations based on search and/or filter conditions.	Application_URL/services/api/v1/search/floc

15 Groups

This API allows you to create, update, retrieve, delete a group and assign, retrieve business objects which are associated with the group.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles GROUP_READ, GROUP_EDIT or GROUP_DELETE assigned to your user ID.
- **API hub link:** [Groups API](#)

Methods

Group	Method	Actions	URI
Business Object Associations	POST	Assigns a Business Object to multiple groups	Application_URL/services/api/v1/businessobjects/{businessObjectId}/assign
	GET	Retrieves all the valid business objects for a group	Application_URL/services/api/v1/groups/type/{groupId}/businessobjects
	POST	Adds business objects to a group	Application_URL/services/api/v1/groups/{groupId}/assign
	GET	Retrieves all the valid business objects for a group	Application_URL/services/api/v1/groups/{groupId}/businessobjects
	PUT	Deletes business objects	Application_URL/services/api/v1/groups/{groupId}/dissociate
	PUT	Removes a business Object from multiple groups	Application_URL/services/api/v1/objects/{businessObjectId}
	GET	Retrieves valid groups for a business object	Application_URL/services/api/v1/objects/{businessObjectId}/groups
List and Header	GET	Retrieves valid group types	Application_URL/services/api/v1/enumeration
	DELETE	Deletes groups	Application_URL/services/api/v1/groups

Group	Method	Actions	URI
	GET	Retrieves all the groups	Application_URL/ services/api/v1/groups
	POST	Creates a group	Application_URL/ services/api/v1/groups
	PUT	Edits group details	Application_URL/ services/api/v1/groups
	PUT	Publish groups	Application_URL/ services/api/v1/groups/ publish
	GET	Retrieves all the information about the group	Application_URL/ services/api/v1/groups/ {groupId}
	POST	Copies a group	Application_URL/ services/api/v1/groups/ {groupId}/copy
	PUT	Creates a new revision	Application_URL/ services/api/v1/groups/ {groupId}/revise

15.1 Additional Information

The following table provides information about the fields of the request payload, along with the corresponding group types, codes, their relevant descriptions and business objects that can be assigned to each group type:

1. Group Type: There can be various group types.

Group Type Code	Description
VAR	Variant
CA	Criticality Assessment
FMEA	FMEA
FLEET	Fleet
GEN	Training
ORG	Organization
PRO	Project
HC	Handover and Commissioning

- BusinessObjectTypes: All 8 business object types are supported in groups. The codes and types of business objects that are being used in the payload:

Business Object Type Code	Description
EQU	Equipment
FL	Location
PRT	Spare Part
MOD	Model
CAT	Template
INS	Instruction
ANN	Announcement
DOC	Documents


- Group Type and Business Objects Mapping: Specific business Object types can be a part of a group type. The mapping is as follows:

Group Type	Business Objects
Criticality Assessment	Equipment, Locations
FMEA	Equipment, Locations
Handover and Commissioning	All - Equipment, Locations, Documents, Instructions, Models, Spare Parts, Announcement, Templates
Training	All - Equipment, Locations, Documents, Instructions, Models, Spare Parts, Announcement, Templates
Project	All - Equipment, Locations, Documents, Instructions, Models, Spare Parts, Announcement, Templates
Organization	All - Equipment, Locations, Documents, Instructions, Models, Spare Parts, Announcement, Templates
Fleet	Equipment
Variant	Model, Spare Parts
Organization	Equipment, Locations

16 Improvement Requests

When in a problem with the product or process, trigger a workflow that brings together the relevant stakeholders of your asset. These stakeholders collaborate and help you resolve your case by giving a solution or recommendation.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles IMPROVEMENT_READ, IMPROVEMENT_EDIT, or IMPROVEMENT_DELETE assigned to your user ID.
- **API hub link:** [Improvement Request API](#) 

Methods

Method	Actions	URI
POST	Creates an improvement request	Application_URL/services/api/v1/improvementrequests
GET	Lists improvement requests based on filter query parameter	Application_URL/services/api/v1/improvementrequests
GET	Returns the count of improvement requests based on filter query parameter	Application_URL/services/api/v1/improvementrequests/\$count
GET	Lists all the configured business partners as providers	Application_URL/services/api/v1/improvementrequests/providers
GET	Lists all the configured business partners as reviewers	Application_URL/services/api/v1/improvementrequests/reviewers
PUT	Update status of the specific improvement request	Application_URL/services/api/v1/improvementrequests/{improvementRequestId}/action/{actionId}
PUT	Update details of the specific improvement request	Application_URL/services/api/v1/improvementrequests/{improvementRequestId}
GET	Gets basic details of an improvement request for a given ID	Application_URL/services/api/v1/improvementrequests/{improvementRequestId}
DELETE	Deletes the specific improvement request	Application_URL/services/api/v1/improvementrequests/{improvementRequestId}
GET	Lists equipment that are assigned to an improvement request	Application_URL/services/api/v1/improvementrequests/{improvementRequestId}/equipments

Method	Actions	URI
GET	Lists documents that are assigned to an improvement request	Application_URL/services/api/v1/improvementrequests/{improvementRequestId}/documents
POST	Post a comment on the improvement request	Application_URL/services/api/v1/improvementrequests/{improvementRequestId}/comments
DELETE	Deletes the comment for a given improvement request	Application_URL/services/api/v1/improvementrequests/{improvementRequestId}/comments/{commentId}

16.1 Additional Information

Request Payload Information

Field Name	Data Type	Description and Possible Values
caseName (mandatory)	String	Specifies the name of the improvement request.
caseType (mandatory)	String	Specifies the improvement request. Possible values are: 1-Product Improvement 2-Process improvement 3-Maintenance improvement
priority (mandatory)	String	Specifies the case priority. Possible values are: 1-Low 2-Medium 3-High
commercialImpact (optional)	Integer	Commercial risk involved due to the performance issue. Possible values: 0 to 5
safetyImpact	Integer	Safety risk. Possible values: 0 to 5
environmentalImpact	Integer	Possible values: 0 to 5
operationalImpact	Integer	Possible values: 0 to 5
urgency	Integer	Possible values: 0 to 5

Field Name	Data Type	Description and Possible Values
referenceObjectId (mandatory)	String	Specifies the affected Object ID
id (mandatory)	String	Specifies the ID of the improvement case that has to be addressed
finalRecommendation(mandatory/ optional based on the improvement case status)	String	Specify a recommendation. It is mandatory when the improvement case is in the Submitted or In Process states.
requestDescription	String	Specify the nature of the problem.
equipmentList	List of JSON Objects	<p>List of equipment IDs that have a problem.</p> <p>assigneeld: Equipment ID</p> <p>isAdded: Flag to determine if equipment is being added or deleted.</p> <p>If true then equipment was added to case, if false the equipment was deleted from the case.</p>

i Note

- A requestor can add/delete when the case is in Created, Author Action, Recommendation Provided, or Rejected states.
- A reviewer can add/delete when the case is in Author Action, Recommendation Provided, Rejected, or In Review states.

Field Name	Data Type	Description and Possible Values
instructionList	List of JSON Objects	<p>List of instruction IDs that are associated with this improvement case.</p> <p>assigneeId: Instruction ID</p> <p>isAdded: Flag to determine if instruction is being added or deleted.</p> <p>If true, then the instruction is being added to the case. If false, then the instruction is being deleted from the case.</p> <p>assigneeType: Determines the section of this instruction that is being added.</p> <p>10: If Request section</p> <p>11: If Recommendation section</p> <div style="border: 1px solid #0070C0; padding: 10px; background-color: #E6E6FA;"> <p>i Note</p> <ul style="list-style-type: none"> • A requestor can add/delete in the Request section when the case is in Created, Author Action, Recommendation Provided, or Rejected states. • A reviewer can add/delete a Request section when the case is in Author Action, Recommendation Provided, Rejected, or In Review states. • A recommendation provider can add/delete the Recommendation section when the case is in In Process or Submitted states. </div>

Field Name	Data Type	Description and Possible Values
documentList	List of JSON Objects	<p>List of document IDs that are associated with this improvement case.</p> <p>assigneeId: Improvement case ID</p> <p>documentID: ID of the document you want to assign</p> <p>assignUnassignFlag: Flag to determine if the document is being added or deleted.</p> <p>If true, then the document is being added to the case. If false, then the document is being deleted from the case.</p> <p>assigneeType: Determines the section to this document that is being added.</p> <p>10: If Request section</p> <p>11: If Recommendation section</p> <div style="border: 1px solid #0070C0; padding: 5px; margin-top: 10px;"> <p>i Note</p> <ul style="list-style-type: none"> • A requestor can add/delete in the Request section when the case is in Created, Author Action, Recommendation Provided, or Rejected states. • A reviewer can add/delete a Request section when the case is in Author Action, Recommendation Provided, Rejected, or In Review states. • A recommendation provider can add/delete the Recommendation section when the case is in In Process or Submitted states. </div>
requestorId (mandatory)	String	ID of the Improvement requestor.
reviewerId (mandatory)	String	ID of the Improvement Reviewer.
processorId (mandatory)	String	ID of the Recommendation Provider

Field Name	Data Type	Description and Possible Values
requestType	String	<p>Specifies the type of the Request</p> <div style="background-color: #f0f0f0; padding: 5px;"> <p>i Note</p> <p>For Improvement Request, it will always be 1.</p> </div>
dueDate	String	<p>Specifies the approximate date by which the request needs to be handled</p>
referenceObjectType (mandatory)	String	<p>Specifies the type of affected object</p> <div style="background-color: #f0f0f0; padding: 5px;"> <p>i Note</p> <p>Following are types supported:</p> <ul style="list-style-type: none"> • MOD : Model • EQU : Equipment • FL : Function Location </div>

17 Instructions


An instruction is a set of steps that help you to carry out a specific task. In SAP Asset Intelligence Network SAP Predictive Asset Insights, a manufacturer shares instructions with an operator. The instructions help the operator to get the most from an item of equipment.

SAP Asset Intelligence Network allows you to create the following types of instructions:

- Planned maintenance
- Breakdown
- Installation
- Operations
- Disposal

For simplicity, the following sections deal with a planned maintenance instruction.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions** SAP Predictive Maintenance and Service, on-premise: You must have the roles INSTRUCTION_READ, INSTRUCTION_EDIT, or INSTRUCTION_DELETE assigned.
- **API hub link:** [Instructions API](#) 

Methods

Method	Actions	URI
DELETE	Delete instruction	Application_URL/services/api/v1/instructions
GET	Retrieves instructions	Application_URL/services/api/v1/instructions
POST	Creates an instruction	Application_URL/services/api/v1/instructions
PUT	Update and returns instruction	Application_URL/services/api/v1/instructions
GET	Returns instruction details for a given instruction ID	Application_URL/services/api/v1/instructions/{instructionId}
GET	Returns announcements assigned to an instruction	Application_URL/services/api/v1/instructions/{instructionId}/announcements
GET	Retrieves all equipments assigned to an instruction	Application_URL/services/api/v1/instructions/{instructionId}/equipments

Method	Actions	URI
GET	Retrieves all models assigned to an instruction	Application_URL/services/api/v1/instructions({instructionId})/models
GET	Returns the number of instructions	Application_URL/services/api/v1/instructions/\$count
GET	Returns enumerations value	Application_URL/services/api/v1/instructions/enumerations
POST	Assign equipment to an instruction	Application_URL/services/api/v1/instructions/equipments/assign
POST	Assign model to instruction	Application_URL/services/api/v1/instructions/models/assign
POST	Assign functional location to instruction	Application_URL/services/api/v1/instructions/assign/floc
PUT	Publish instructions	Application_URL/services/api/v1/instructions/publish

17.1 Additional Information

The following table provides information about the fields of the request payload, along with the corresponding data types, and the data type descriptions:

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
instructionType (Mandatory)	String	Specifies the type of instruction. You can create five types of instructions. The following list provides the types of instructions and the corresponding codes: <ul style="list-style-type: none"> Planned Maintenance - 1 Breakdown - 2 Installation - 3 Operations - 4 Disposal - 5

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
subClass (Mandatory)	String	Specifies the subclass that the instruction belongs to.
		<p>i Note</p> <p>If you want to assign the instruction to a model, the model must belong to the same subclass as that of an instruction.</p>
status (Mandatory)	String	Specifies the status of an instruction. The following list provides the list of statuses and the corresponding status codes: <ul style="list-style-type: none"> • Unpublished - 1 • Published - 2 • In revision - 3
activityCode (Mandatory)	String	Specifies an activity code for a primary action performed as part of the instruction. For more information, see the Activity Codes table.
Title (Mandatory)	String	Specifies the title of an instruction.
instructionDescription (Optional)	String	Specifies a detailed description for an instruction.
duration (Mandatory)	Double	Specifies the expected time required to execute an instruction. You can use this field when you create a planned maintenance instruction.
durationUnit (Mandatory)	String	Specifies the unit of time for the duration. You can use this field when you create a planned maintenance instruction. For more information, see the Duration Unit and Codes table.
peopleRequired (Optional)	Integer	Specifies the number of people required to execute the instruction.

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
documentID in documentIDList at the instructionHeader level (Optional)	String	<p>Specifies the document ID that you can assign at the instruction header level.</p> <p>For a list of supported MIME types, see the Supported MIME Types for Attachments table.</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;"> <p>i Note</p> <p>The document ID is a unique ID generated by the system when you upload a document.</p> </div>
documentID in vdsDocumentIDList at the instructionHeader level (Optional)	String	<p>Specifies the document ID of a 3D visual file that you want to assign at an instruction header level.</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;"> <p>i Note</p> <p>The document ID is a unique ID generated by the system when you upload a document.</p> </div>
documentID in documentIDList at the steps level (Optional)	String	<p>Specifies the document ID of an image file or a 3D visual that you want to assign at an instruction header level.</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;"> <p>i Note</p> <p>The document ID is a unique ID generated by the system when you upload a document.</p> </div>
documentID at the steps level (Optional)	String	<p>Specifies the document ID that you want to assign to a step.</p> <p>For a list of supported mime types, see Table Supported MIME Types for Attachments.</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;"> <p>i Note</p> <p>The document ID is a unique ID generated by the system when you upload a document.</p> </div>

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
failureCode (Mandatory)	String	<p>Specifies failure codes that define different methods in which an item of equipment, or a model might break-down. These codes are based on ISO standards and you can use them when you create a breakdown instruction.</p> <p>For more information on various failure mode codes, see the Failure Codes table.</p>
frequency (Mandatory)	Integer	Specifies the frequency at which the instruction must be executed. You can use this field when you create a planned maintenance instruction.
frequencyUnit (Mandatory)	String	Specifies the unit of time for the frequency
criticality (Mandatory)	String	Specifies how important an instruction is for the general maintenance and reliability of an item of equipment. You can use this field when you create a planned maintenance instruction.
postCheckDescription (Optional)	String	Specifies the post checks to be performed after executing an instruction.
ruleCode (Optional)	String	<p>Specifies the safety rule code that you can assign to an instruction.</p> <p>For more information, see the Safety Rule Codes table.</p>
stepName (Mandatory)	String	Specifies the name of a step in an instruction.
stepDescription (Optional)	String	Specifies the step description for an instruction.

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
riskCategory (Optional)	String	<p>Specifies the level of potential harm to the environment or injury to the user when you carry out a step in an instruction. The following list contains the list of risk categories and the corresponding codes:</p> <ul style="list-style-type: none"> • Low - 1 • Medium - 2 • High - 3
riskDescription(Optional)	String	Specifies a risk description.
roleCode (Optional)	String	<p>Specifies the role code that corresponds to the role that you require to execute a step in an instruction.</p> <p>For more information, see the Role Codes table.</p>
sparePartID (Optional)	String	Specifies the spare part ID.
quantity (Optional)	Double	Specifies the quantity for a spare part.
toolName (Optional)	String	Specifies the name of the tool required to perform an instruction.
modelID (optional)	String	Specifies the ID of the model assigned to the instruction.
		<div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p>i Note</p> <p>The model ID is a unique ID generated by the system when you create a model.</p> </div>
updateDeleteFlag	String	<p>Specifies if you can update or delete the steps, preconditions, attachments, or post checks in an instruction.</p> <p>Possible values include 'u' for update and 'd' for delete.</p>

Field Name (Mandatory or Optional)	Data Type	Description and Possible Values
assignUnassignFlag	String	Specifies whether you can assign a model to an instruction or remove an assignment of a model from an instruction. Also, specifies whether you can assign an attachment to a step in an instruction, or remove an assignment of a step in an instruction. Possible values include 'a' for assign and 'ua' for removing an assignment.

Duration Unit and Codes

Field	Code	Description
durationUnit	1	Minutes
	2	Hours
	3	Days
	4	Weeks
	5	Months
	6	Years

Role Codes

Field	Code	Description
roleCode	1	Heavy Equipment Technician
	2	Maintenance Technician
	3	Electrical Technician
	4	Electric Line Worker
	5	Maintenance Helper
	6	Welder-Fitter
	7	Diver
	8	Crane Operator
	9	Heavy Equipment Operator
	10	Health and Safety Advisor

Field	Code	Description
	11	Instrument Technician
	12	Maintenance Supervisor
	13	Marine Engineer
	14	Painter/Coater
	15	Pipe Fitter
	16	Scaffolder
	17	Pipeline Inspector
	18	Mechanical Inspector
	19	Electrical Inspector
	20	Instrumentation Inspector

Safety Rule Codes

Field	Code	Description
ruleCode	1	Wear a safety helmet
	2	Read the safety manual
	3	Wear safety overalls
	4	Ground connection is required
	5	Wear ear protection
	6	Wear eye protection
	7	Wear safety boots
	8	Wear safety gloves

Failure Codes


Field	Code	Description
failureCode	FTF	Fail to function
	FTO	Fail to open
	FTC	Fail to close
	DOP	Operation delay

Field	Code	Description
	HIO	Output is high
	LOO	Output is low

18 Locations

This API allows you to perform CRUD operations on a location that corresponds to a virtual record of the location where an equipment is installed. The API also allows you to manage templates, attributes, attachments and tags for a location.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles FUNCTIONAL_LOCATION_READ, FUNCTIONAL_LOCATION_EDIT or FUNCTIONAL_LOCATION_DELETE assigned.
- **API hub link:** [Location API](#) 

Methods

Group	Method	Actions	URI
List and Header	GET	Lists locations based on filter query parameter	Application_URL/services/api/v1/location
	POST	Creates a location	Application_URL/services/api/v1/location
	PUT	Updates location	Application_URL/services/api/v1/location
	GET	Retrieves a set of locations	Application_URL/services/api/v1/location/internalid/{internalid}/header
	DELETE	Deletes a location with specific ID	Application_URL/services/api/v1/location/{id}
	GET	Retrieves a specific location	Application_URL/services/api/v1/location/{id}
	GET	Retrieves installation location	Application_URL/services/api/v1/location/{id}/installationlocation
	PUT	Updates installation location	Application_URL/services/api/v1/location/{id}/installationlocation
	GET	Retrieves locations count	Application_URL/services/api/v1/location/\$count

Group	Method	Actions	URI
Attachments	GET	Retrieves attachments of a location	Application_URL/ services/api/v1/location/{id}/attachments
	POST	Assigns attachments to specific location	Application_URL/ services/api/v1/location/{id}/attachments
	GET	Retrieves tags of a location	Application_URL/ services/api/v1/location/{id}/tags
	PUT	Updates tags of a location	Application_URL/ services/api/v1/location/{id}/tags
Component Structure	GET	Retrieves components for a location	Application_URL/ services/api/v1/location/{id}/components
	PUT	Updates component structure of the location	Application_URL/ services/api/v1/location/{id}/components
	PATCH	Updates equipment of a location	Application_URL/ services/api/v1/location/{id}/equipment
Lifecycle	PUT	Updates status of a location to Publish	Application_URL/ services/api/v1/location/{id}/publish
	PUT	Updates status of a location to InRevision	Application_URL/ services/api/v1/location/{id}/revise
	PUT	Updates status of locations to Publish	Application_URL/ services/api/v1/locations/publish
Attribute Values	GET	Retrieves templates of a location	Application_URL/ services/api/v1/location/{id}/templates
	GET	Retrieves attributes of a location	Application_URL/ services/api/v1/location/{id}/values
	PUT	Updates attributes of a location	Application_URL/ services/api/v1/location/{id}/values
	PUT	Updates location templates	Application_URL/ services/api/v1/location/templates

19 Models

A model is an abstract representation from the manufacturer that defines all maintenance and specification information related to a new or existing product. A physical equipment is an instance of a model.

A model comprises of the following information:

- **Header:** Primary information that describes a model such as name, description, to name a few.
- **Attributes:** Properties of a model.
- **Components:** Detachable objects of a model.
- **Assignments:** Objects that are associated with a model such as instructions, documents, to name a few.

Public APIs are available for each of these specific sections of a model.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles MODEL_READ, MODEL_EDIT, and MODEL_DELETE assigned to your user ID.
- **API hub link:** [Model API](#)

Methods

Group	Method	Actions	URI
Model List and Header	POST	Creates an equipment model	Application_URL/ services/api/v1/models
	GET	Retrieves all equipment models	Application_URL/ services/api/v1/models
	GET	Retrieves the number of models based on the filter query	Application_URL/ services/api/v1/models/ \$count
	GET	Retrieves list model headers for the specified model name	Application_URL/ services/api/v1/models/ internalid/{name}/header
	GET	Retrieves model header for the specified model	Application_URL/ services/api/v1/ models({modelId})/header
	PUT	Updates a model header	Application_URL/ services/api/v1/ models({modelId})/header
Model Lifecycle	DELETE	Deletes the model	Application_URL/ services/api/v1/ models({modelId})

Group	Method	Actions	URI
	PUT	Publishes multiple models	Application_URL/ services/api/v1/models/ publish
	PUT	Creates revision of a model	Application_URL/ services/api/v1/ models({modelId})/revise
	PUT	Publishes a model in the Draft or the Revision state	Application_URL/ services/api/v1/ models({modelId})/publish
Model Values	GET	Retrieves the attribute values of a model	Application_URL/ services/api/v1/ models({modelId})/values
	PUT	Updates attribute values of a model	Application_URL/ services/api/v1/ models({modelId})/values
	PATCH	Updates attribute values of a model	Application_URL/ services/api/v1/ models({modelId})/values
	GET	Retrieves templates associated with the model	Application_URL/ services/api/v1/ models({modelId})/ templates
	PUT	Modifies the template associations of a model	Application_URL/ services/api/v1/ models({modelId})/ templates
Model Object Associations	GET	Retrieves all documents assigned to a model	Application_URL/ services/api/v1/ models({modelId})/documents
	POST	Assigns documents to a model	Application_URL/ services/api/v1/ models({modelId})/documents
	GET	Lists instructions of a model	Application_URL/ services/api/v1/ models({modelId})/instructions
	PUT	Modifies the instruction list of models	Application_URL/ services/api/v1/models/ instructions
	GET	Get announcements assigned to a model	Application_URL/ services/api/v1/ models({modelId})/ announcements

Group	Method	Actions	URI
	GET	Retrieves spare parts assigned to a model	Application_URL/services/api/v1/models({modelId})/spareparts
	PUT	Modifies sparepart list of a model	Application_URL/services/api/v1/models/spareparts
Other	GET	Retrieves timeline of a model	Application_URL/services/api/v1/models({modelId})/timeline
	GET	Get all organizations in the system	Application_URL/services/api/v1/organizations
	GET	Get organizations with specific role	Application_URL/services/api/v1/organizations/byrole
	GET	Get the count of all organizations	Application_URL/services/api/v1/organizations/\$count
	GET	Retrieves enumeration values	Application_URL/services/api/v1/enumeration
	GET	Retrieves value of an enumeration type	Application_URL/services/api/v1/enumerationvalue
	GET	Search for a model based on its attributes values	Application_URL/services/api/v1/models/attributeSearch
Model Structure	GET	Retrieves components attached to a model	Application_URL/services/api/v1/models({modelId})/components
	PUT	Updates the components associated with a model	Application_URL/services/api/v1/models({modelId})/components
	PATCH	Updates component structure of the model	Application_URL/services/api/v1/models({modelId})/components
	GET	Retrieves a list of immediate model parents to a model	Application_URL/services/api/v1/models({modelId})/allparents
Model Search	PUT	search for available models based on search and/or filter conditions	Application_URL/services/api/v1/search/model

Group	Method	Actions	URI
Model Indicators	POST	Configures the threshold	Application_URL/ services/api/v1/model({id})/ indicatorconfiguration
	GET	Retrieves an indicatorconfi- guration	Application_URL/ services/api/v1/indicator- configuration
	PUT	Modify indicatorconfigura- tion	Application_URL/ services/api/v1/indicator- configuration({id})
	POST	Modify indicator values	Application_URL/ services/api/v1/indicatorval- ues
	GET	Retrieves indicator values	Application_URL/ services/api/v1/indicatorval- ues
	PUT	Updates the new value to an indicator	Application_URL/ services/api/v1/indicator- value({id})
	DELETE	Deletes indicator	Application_URL/ services/api/v1/indicator- value({id})
	GET	Retrieve indicator values of a model	Application_URL/ services/api/v1/model({id})/ indicatorvalues
	PUT	Resets the indicator value	Application_URL/ services/api/v1/ model({modelId})/reset/ indicatorvalues
	Model Successor-Predeces- sor	GET	Retrieves the list of succes- sors for the given model
GET		Retrieves the list of assigned successors for the given model	Application_URL/ services/api/v1/model({id})/ predecessors
GET		Retrieves the list of prede- cessors for the given model	Application_URL/ services/api/v1/ model({modelId})/reset/ indicatorvalues
POST		Assign successor(s) to the model	Application_URL/ services/api/v1/model({id})/ successor/assign
PUT		Un-assign successor(s) from the model	Application_URL/ services/api/v1/model({id})/ successor/unassign

Group	Method	Actions	URI
Nested Structure Indicator	GET	Get nested structure indicator metadata.	Application_URL/services/api/v1/model({modelId})/nested/structure/indicator
	POST	Post nested structure indicator values.	Application_URL/services/api/v1/nested/structure/indicatorvalues
	POST	Get nested structure indicator values.	Application_URL/services/api/v1/nested/structure/read/indicatorvalues

19.1 Model Search

Constructing Search Request

Free-Text Search

You can search for models based on the following model fields: **internalId**, **longDescription**, **shortDescription**, **gtin**, **generation**.

Example: If a model has an internal ID: ABC_MOD, the search-text **ABC_**, **A**, **AB**, **ABC_M**, and so on, will return the model details of **ABC_MOD** in the response.

Sample Request:

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": "ABC_",
  "orderBy": {
    "path": "changedOn",
    "descending": true
  },
  "select": [],
  "attributes": []
}
```

Sorting Search Results

The search results can be sorted by defining the 'orderBy' condition. Supported model fields for orderBy are: **internalId**, **changedOn**, **shortDescription**, **subclassName**, **manufacturer**, **completeness**, **createdOn**, **publishedOn**, **source**, **className**.

i Note

Default Value for 'orderBy' will be sorting based on **changedOn** in the descending order.

orderBy definition takes structure:

Sample Code

```
{
  "path": "<model_header_field>",
  "descending": <true/false>
}
```

Constructing Filters for Search

- **Basic Filter Structure:**

Filters are used to define the search conditions. A basic filter can be defined as:

Sample Code

```
{
  "operator": "<operator>",
  "value1": "<value>",
  "value2": "<value>", # Required when defining a range
  "path": "<field>"
}
```

Filter conditions support the following:

- Operators
 - "EQ" - Equals
 - "GT" - Greater than
 - "GE" - Greater than or Equal to
 - "LT" - Less than
 - "LE" - Less than or Equal to
 - "IN" - Present in a list/array
 - "BT" - Between a range (range is defined by setting value1 & value2 fields)
- Fields: - **internalId, modelId, changedOn, shortDescription, subclassName, completeness, createdOn, publishedOn, source, status, classId, className, serviceExpirationDate, createdOn, hasInRevision, publishedOn, version, templates, primaryTemplateId, primaryTemplateName, subclassId, subclassName, descriptions, releaseDate, generation, image, completeness, gtin, status, source, sourceId, manufacturerId.**

i Note

The filter value defined in "value1"/"value2" is not free text enabled. The exact term must be entered for the search to return the desired results.

Examples:

1. Filter for an attribute.

☰ Sample Code

```
{
  "operator": "EQ",
  "value1": "D591D46E4C26451B9FC39765954F492C",
  "path": "attributeId"
}
```

2. Filter all models published between two dates.

☰ Sample Code

```
{
  "operator": "BT",
  "value1": 1592213215896,
  "value2": 1608024415896,
  "path": "publishedOn"
}
```

3. Filter for models having primary template as T1 or T2.

☰ Sample Code

```
{
  "operator": "IN",
  "value1": ["T1","T2"],
  "path": "primaryTemplateName"
}
```

- **Combining Multiple Filters:**

Multiple filters can be combined to form multiple or nested filters.

- Multiple filters are formed when more than one filter is defined in a level connected by a logical operation (Logical AND or Logical OR).
- Nested filters are formed when a hierarchy of filters is defined at multiple level connected by a logical operation (Logical AND or Logical OR).

The basic structure is as follows:

☰ Sample Code

```
{
  "filters": [{
    "operator": "<operator>",
    "value1": "<value>",
    "path": "<path>"
  },
  {
    "operator": "<operator>",
    "value1": "<value>",
    "path": "< path>"
  }
],
  "and": true. # Logical AND/OR
}
```

Example:

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": null,
  "orderBy": {
    "path": "changedOn",
    "descending": true
  },
  "filter": {
    "filters": [
      {
        "filters": [
          {
            "operator": "EQ",
            "value1": "0E261150A7AA4934BE493D020E54C18C",
            "path": "attributeId"
          },
          {
            "operator": "EQ",
            "value1": "true",
            "path": "value"
          }
        ],
        "and": true
      },
      {
        "filters": [
          {
            "operator": "BT",
            "value1": 1606258801000,
            "value2": 1606345199000,
            "path": "changedOn"
          }
        ]
      }
    ],
    "and": false
  }
}
```

Pagination

Pagination can be achieved by simultaneously setting the top and skip request options with every request.

i Note

The default value of top and skip will be 20 and 0.

The supported maximum value of top + skip is 1000.

Pagination using SearchAfter

For pagination in AWS beyond 1000 objects, use the property **searchAfter**. This value will be the part of every response and must be sent in the subsequent call to get the next batch of records.

i Note

- The **orderBy** field won't be applied in case **searchAfter** is being used.
- The skip parameter won't be applied if **searchAfter** is being used.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": "ABC ",
  "orderBy": null,
  "select": [],
  "attributes": [],
  "searchAfter": ""
}
```

Example:

Sample Code

```
{
  "count": 0,
  "value": [],
  "searchAfter": "4BC02F4450CE4321B3112937346944C2"
}
```

Response Fields Selection

The 'select' parameter in the request body determines the fields to be present in the response.

The supported fields are : **modelId**, **internalId**, **status**, **classId**, **className**, **serviceExpirationDate**, **createdOn**, **hasInRevision**, **publishedOn**, **version**, **templates**, **primaryTemplateId**, **primaryTemplateName**, **subclassId**, **subclassName**, **descriptions**, **releaseDate**, **generation**, **image**, **completeness**, **gtin**, **status**, **source**, **sourceId**, **manufacturerId**, **manufacturer**, **type**, **attributes**.

If 'select' is not defined in the request, the response will contain only the fields **modelId** and **internalId**.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "select": ["modelId", "internalId", "status", "classId", "subclassName"],
  "attributes": []
}
```

Filtering Attributes

The 'attributes' parameter in the request body defines the attributes to be considered for the search operation. It takes a list of valid attribute IDs for which the user has authorization to access. Any attributes which are not mentioned under the 'attributes' section will not be part of the response.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "select": ["modelId", "internalId", "status", "classId", "subclassName"],
  "attributes":
  ["21858B573E3A49A9AA93F82521B0D13D", "32969C64F4B49A9AA93F82521B0D13D"]
}
```

Example Scenarios

1. Searching with attribute values

Scenario: Search for all models having an attribute with ID '21858B573E3A49A9AA93F82521B0D13D' and value as '23'.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": null,
  "orderBy": {
    "path": "changedOn",
    "descending": true,
    "type": null
  },
  "attributes": [
    "21858B573E3A49A9AA93F82521B0D13D"
  ],
  "filter": {
    "filters": [
      {
        "filters": [
          {
            "operator": "EQ",
            "value1": "21858B573E3A49A9AA93F82521B0D13D",
            "path": "attributeId"
          },
          {
            "operator": "EQ",
            "value1": "23",
            "path": "value1"
          }
        ],
        "and": true
      }
    ],
    "and": true
  }
}
```

2. Free-text search combined with multiple filters

Scenario: Search for all models with an internal ID that starts with "abc"; and the last changed date between two dates and attribute ID as '9DC51C442950460EB0AD947A7A6DD05E'.

Sample Code

```
{
  "top": 20,
  "skip": 0,
  "search": "abc",
  "orderBy": {
    "path": "changedOn",
    "descending": true,
    "type": null
  },
  "attributes": [ "9DC51C442950460EB0AD947A7A6DD05E" ],
  "filter": {
    "filters": [
      {
        "filters": [
          {
            "operator": "EQ",
```

```
        "value1": "0E261150A7AA4934BE493D020E54C18C",
        "path": "attributeId"
      }
    ],
  },
  {
    "filters": [
      {
        "operator": "BT",
        "value1": 1606258801000,
        "value2": 1606345199000,
        "path": "changedOn"
      }
    ]
  }
],
"and": false
}
}
```


Limitations

- This API is only available on SAP BTP, Cloud Foundry environment (Amazon Web Services (AWS)).
- Filter on attributes of type picture, enum, attributes with codelists of type string, date, numeric, and numeric flexible having multiselect option and along with unit of measure are not supported.
- You cannot use this function to search for models using the unit of measure (UoM) assigned to the attribute value.
- The search results are 'near real-time', there might be a slight delay between when the changes are made and when they are available in the search results.
- Partial text search on string attributes is not supported.

20 Model Requests

This request triggers a workflow that brings together the relevant stakeholders of your equipment. These stakeholders collaborate and help you to find out the relevant model recommendation for equipment.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles MODEL_REQUEST_READ, MODEL_REQUEST_EDIT, or MODEL_REQUEST_DELETE assigned to your user ID.
- **API hub link:** [Model Request API](#) 

Methods


Method	Actions	URI
POST	Creates a model request	Application_URL/services/api/v1/modelrequests
GET	Lists model requests based on filter query parameter	Application_URL/services/api/v1/modelrequests
GET	Returns the count of model requests based on filter query parameter	Application_URL/services/api/v1/modelrequests/\$count
GET	Lists all the configured business partners as providers	Application_URL/services/api/v1/modelrequests/providers
PUT	Update status of the specific model request	Application_URL/services/api/v1/modelrequests/{modelrequestId}/action/{actionId}
PUT	Update details of the specific model request	Application_URL/services/api/v1/modelrequests/{modelrequestId}
GET	Gets basic details of a model request	Application_URL/services/api/v1/modelrequests/{modelrequestId}
DELETE	Deletes the specific model request	Application_URL/services/api/v1/modelrequests/{modelrequestId}
GET	Lists equipment that are assigned to a model request	Application_URL/services/api/v1/modelrequests/{modelrequestId}/equipments
GET	Lists models that are assigned to a model request	Application_URL/services/api/v1/modelrequests/{modelrequestId}/models
GET	Lists documents that are assigned to a model request	Application_URL/services/api/v1/modelrequests/{modelrequestId}/documents

Method	Actions	URI
POST	Post a comment on the model request	Application_URL/services/api/v1/ modelrequests/{modelrequestId}/ comments
DELETE	Deletes the comment for a given model request	Application_URL/services/api/v1/ modelrequests/{modelrequestId}/ comments/{commentId}

21 Network Authorizations

Network authorization allows creation of authorization groups for business partners and provides them with access to business objects. You can grant permissions such as read or write access to business objects within the group.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles SHARING_READ, SHARING_EDIT, or SHARING_DELETE assigned to your user ID.
- **API hub link:** [Network Authorization API](#) 

Methods

Group	Method	Actions	URI
Dependent Objects	GET	Returns dependent object types	Application_URL/services/api/v1/authorization/dependentobjecttypes
Authorization Groups	GET	Returns authorization groups	Application_URL/services/api/v1/authorization/groups
	POST	Creates authorization group	Application_URL/services/api/v1/authorization/groups
	GET	Counts authorization groups	Application_URL/services/api/v1/authorization/groups/\$count
	DELETE	Deletes authorization group	Application_URL/services/api/v1/authorization/groups({groupId})
	GET	Returns header of an authorization group	Application_URL/services/api/v1/authorization/groups({groupId})/header
	PUT	Updates header of an authorization group	Application_URL/services/api/v1/authorization/groups({groupId})/header
	PUT	Adds or removes business partners	Application_URL/services/api/v1/authorization/groups({groupId})/members

Group	Method	Actions	URI
	GET	Returns the list of objects	Application_URL/ services/api/v1/authoriza- tion/groups({groupId})/ objects
	PUT	Updates objects of an au- thorization group	Application_URL/ services/api/v1/authoriza- tion/groups({groupId})/ objects
	POST	Shares list of depend- ent objects with authoriza- tion group	Application_URL/ services/api/v1/authoriza- tion/groups({groupId})/ objects/dependencies
	PUT	Updates dependent objects	Application_URL/ services/api/v1/authoriza- tion/groups({groupId})/ objects/dependent
Objects	GET	Returns authorization groups	Application_URL/ services/api/v1/groups/ authorization/objects/{id}/ groups
	GET	Returns business partners	Application_URL/ services/api/v1/authoriza- tion/objects/{id}/partners
Business Partners	GET	Returns authorization groups	Application_URL/ services/api/v1/authoriza- tion/partners/{id}/groups
	GET	Returns objects	Application_URL/ services/api/v1/authoriza- tion/partners/{id}/objects

22 Notifications

A notification facilitates the process of maintenance of equipments and other related processes. You can use the API operations to create and manage the notification data.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles EQUIPMENT_EDIT or EQUIPMENT_DELETE for equipment and FUNCTIONAL_LOCATION_EDIT or FUNCTIONAL_LOCATION_DELETE for locations assigned to your user ID.
- **API hub link:** [Notifications API](#)

Methods

Method	Actions	URI
POST	Creates a notification	Application_URL/services/api/v1/notification
PUT	Updates a notification	Application_URL/services/api/v1/notification
GET	Retrieves notifications	Application_URL/services/api/v1/notification
GET	Retrieves the count of notifications	Application_URL/services/api/v1/notification/\$count
GET	Search for notification based on filter query parameter	Application_URL/services/api/v1/notificationswithoutexternalid

Parameters:

Operation

Parameters

notificationCreate

- externalID: string, External ID of notification.
- externalSystemID: string, External System ID of notification.
- equipmentID: string, Equipment ID of notification.
- locationID: string, Location ID of notification.
- breakdown: string, Equipment breakdown ID.
- type*: string
Type of notifications and its meanings M1: Maintenance Request, M2: BreakDown.
Enum: [M1, M2]
- priority*: string
Priorities and its meanings 5: Low, 10: Medium, 15: High, 20: Very High, 25: Emergency.
Enum: [5, 10, 15, 20, 25]
- status*: string
Status types and its meanings NEW: New, PBD: Published, CPT: Completed.
Enum: [NEW, PBD, CPT]
- startDate: string(\$YYYY-MM-DD), Start date of notification
- endDate: string(\$YYYY-MM-DD), End date of notification
- malfunctionStartDate: string(\$YYYY-MM-DDTHH:mm:ss), Start date of malfunction
- malfunctionEndDate: string(\$YYYY-MM-DDTHH:mm:ss), End date of malfunction
- operator: string, Operator ID for notifications
- workCenter: string, maxLength: 40, Work center of notifications
- workCenterSystemId: string, maxLength: 32, Work center system ID for notifications
- maintenancePlannerGroup: string, maxLength: 32, Maintenance plan group of notifications
- maintenancePlanGroupSystemId: string, maxLength: 32, Maintenance plan group system ID of notifications
- proposedFailureModelID: string, Proposed FailureMode ID of notifications
- confirmedFailureModelID: string, Confirmed Failure-Mode ID of notifications
- systemProposedFailureModelID: string, System Failure-Mode ID of notifications
- effectID: string, Effect ID of notifications
- instructionID: string, Instruction ID of notifications
- causeID: string, Cause ID of notifications

-
- description*:
 - description: The description of the notification in logged on language of the user in which description needs to be maintained. This description object will be ignored when the below list of descriptions are provided in the payload. We recommend you to use 'descriptions' to provide list of descriptions, since 'description' would be deprecated with next release.
 - short*: string, maxLength: 255, Short description
 - long: string, maxLength: 5000, Long description
 - descriptions: The list of all descriptions of the notification in different languages in which description needs to be maintained. The API will perform a complete refresh of the list descriptions.
 - languageSoCode: string, maxLength: 2, Language in which description is available.
 - shortDescription: string, maxLength: 255, Short description
 - longDescription: string, maxLength: 5000, Long description
 - geometry:
 - description: Geographical co ordinates of the location of the notification origin equipment
 - type: string, maxLength: 2, Type of geographical location information.
 - coordinates: string, Geographical coordintaes of the location of the notification origin equipment in latitudes and longitudes
 - alertIDs:
 - id*: string, SAP IoT Event(Alert) id.
 - alertType: string, Alert type for corresponding alert.
 - items:
 - itemNumber: string, Notification item number.
 - shortDescription: string, Notification item short description.
 - longDescription: string, Notification item short description.
 - notificationFailureModes:
 - description: Notification failure modes.
 - failureModelID: string, maxLength: 255, Short description
-

Operation

Parameters

- notificationCauseCodes:
 - causeID: String, Notification cause id.
 - causeDescription: string, maxLength: 255, Notification cause id description.
-

Operation

Parameters

notificationUpdate

- notificationID*: string, ID of notification.
- equipmentID: string, Equipment ID of notification.
- locationID: string, Location ID of notification.
- breakdown: string, Equipment breakdown ID.
- type*: string
Type of notifications and its meanings M1: Maintenance Request, M2: BreakDown.
Enum: [M1, M2]
- priority*: string
Priorities and its meanings 5: Low, 10: Medium, 15: High, 20: Very High, 25: Emergency.
Enum: [5, 10, 15, 20, 25]
- status*: string
Status types and its meanings NEW: New, PBD: Published, CPT: Completed.
Enum: [NEW, PBD, CPT]
- startDate: string(\$YYYY-MM-DD), Start date of notification
- endDate: string(\$YYYY-MM-DD), End date of notification
- malfunctionStartDate: string(\$YYYY-MM-DDTHH:mm:ss), Start date of malfunction
- malfunctionEndDate: string(\$YYYY-MM-DDTHH:mm:ss), End date of malfunction
- operator: string, Operator ID for notifications
- workCenter: string, maxLength: 40, Work center of notifications
- workCenterSystemId: string, maxLength: 32, Work center system ID for notifications
- maintenancePlannerGroup: string, maxLength: 32, Maintenance plan group of notifications
- maintenancePlanGroupSystemId: string, maxLength: 32, Maintenance plan group system ID of notifications
- proposedFailureModelID: string, Proposed FailureMode ID of notifications
- confirmedFailureModelID: string, Confirmed Failure-Mode ID of notifications
- systemProposedFailureModelID: string, System Failure-Mode ID of notifications
- effectID: string, Effect ID of notifications
- instructionID: string, Instruction ID of notifications
- causeID: string, Cause ID of notifications
- description*:

-
- description: The description of the notification in logged on language of the user in which description needs to be maintained. This description object will be ignored when the below list of descriptions are provided in the payload. We recommend you to use 'descriptions' to provide list of descriptions, since 'description' would be deprecated with next release.
 - short*: string, maxLength: 255, Short description
 - long: string, maxLength: 5000, Long description
 - descriptions: The list of all descriptions of the notification in different languages in which description needs to be maintained. The API will perform a complete refresh of the list descriptions.
 - languageSoCode: string, maxLength: 2, Language in which description is available.
 - shortDescription: string, maxLength: 255, Short description
 - longDescription: string, maxLength: 5000, Long description
 - geometry:
 - description: Geographical coordinates of the location of the notification origin equipment
 - type: string, maxLength: 2, Type of geographical location information.
 - coordinates: string, Geographical coordinates of the location of the notification origin equipment in latitudes and longitudes
 - alertIDs:
 - id*: string, IOT Leonardo Event(Alert) id.
 - alertType: string, Alert type for corresponding alert.
 - items:
 - itemNumber: string, Notification item number.
 - shortDescription: string, Notification item short description.
 - longDescription: string, Notification item short description.
 - notificationFailureModes:
 - description: Notification failure modes.
 - failureModelID: string, maxLength: 255, Short description
 - notificationCauseCodes:


Operation**Parameters**

- causeID: String, Notification cause id.
 - causeDescription: string, maxLength: 255, Notification cause id description.
-

23 Organizational Structure

Allows you to create and read Organizational Structure

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles EQUIPMENT_EDIT or EQUIPMENT_DELETE for equipment and FUNCTIONAL_LOCATION_EDIT or FUNCTIONAL_LOCATION_DELETE for locations assigned to your user ID.
- **API hub link:** [Organizational Structure API](#) 

Methods

Method	Actions	URI
GET	Get all organizational structure based on filter	Application_URL/services/api/v1/organizationalstructures
POST	Create organizational structure	Application_URL/services/api/v1/organizationalstructures
GET	Get organizational structure based on the salesAreaID	Application_URL/services/api/v1/organizationalstructures/{{getbyid}}
GET	Checks validity of organizational structure based on data provided	Application_URL/services/api/v1/orgstructure({orgid})/isvalidorg

24 Spare Parts

Spare parts are components that are kept in your inventory as spare. You can record a spare part's manufacturing information, dimensions, quantity, shelf life, to name a few. A spare part inherits attributes from a subclass. You can assign a spare part to a model.

Public APIs are available for creating, updating, viewing and deleting spare parts. You can also view steps and models assigned to a part using APIs.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles PART_READ, PART_EDIT, or PART_DELETE assigned to your user ID.
- **API hub link:** [Spare Parts API](#)

Methods


Group	Method	Actions	URI
Parts	GET	Retrieves List of spare part	Application_URL/ services/api/v1/parts
	POST	Creates a spare part	Application_URL/ services/api/v1/parts
	PUT	Updates details of specified spare part	Application_URL/ services/api/v1/parts
	GET	Retrieves spare part header information for specific internalId	Application_URL/ services/api/v1/parts/internalid/{internalid}/header
	GET	Retrieves spare part header information for a Specific spare part	Application_URL/ services/api/v1/parts/{partid}
	PUT	Deletes a spare part	Application_URL/ services/api/v1/parts/ remove
Documents	GET	Retrieves the list of documents associated with the spare parts	Application_URL/ services/api/v1/parts/{partid}/documents
	PUT	Modify documents associated with the spare part	Application_URL/ services/api/v1/parts/documents
Spare Parts Object Associations	GET	Retrieves all the equipment assigned to a spare part	Application_URL/ services/api/v1/parts/{partid}/equipment

Group	Method	Actions	URI
	GET	Retrieves all the models assigned to a spare part	Application_URL/ services/api/v1/ parts({partid})/models
	PUT	Updates the operating status for a spare part	Application_URL/ services/api/v1/ parts({partid})/phase
	GET	Retrieve predecessor for a spare part	Application_URL/ services/api/v1/ parts({partid})/predecessor
	GET	Retrieves all the steps assigned to a spare part	Application_URL/ services/api/v1/ parts({partid})/steps
	GET	Retrieve successors for a spare part	Application_URL/ services/api/v1/ parts({partid})/successors
	PUT	Assign successors to the spare part	Application_URL/ services/api/v1/ parts({partid})/successors
	GET	Retrieves all the templates assigned to a spare part	Application_URL/ services/api/v1/ parts({partid})/templates
	PUT	Assign templates to a spare part	Application_URL/ services/api/v1/ parts({partid})/templates
Attribute Value	GET	Retrieve attribute values for the spare part	Application_URL/ services/api/v1/ parts({partid})/values
	PUT	Modify attribute values of the spare part	Application_URL/ services/api/v1/ parts({partid})/values
Others	GET	Retrieves the enumeration Values	Application_URL/ services/api/v1/parts/ enumerations

25 Systems

Systems enable you to manage a composition of equipment together which have a certain function to perform.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles SYSTEM_READ, SYSTEM_EDIT, or SYSTEM_DELETE assigned to your user ID.
- **API hub link:** [Systems API](#) 

Methods

Method	Actions	URI
GET	Lists systems based on filter query parameter	Application_URL/services/api/v1/systems
POST	System Creation	Application_URL/services/api/v1/systems
GET	All authorized systems count	Application_URL/services/api/v1/systems/\$count
GET	Retrieves system header information for the specific system	Application_URL/services/api/v1/systems(systemId)/header
PUT	Edit a system	Application_URL/services/api/v1/systems(systemId)/header
GET	Retrieves list of systems headers for the specified systems internalId	Application_URL/services/api/v1/systems/internalid/{internalId}/header
DELETE	Delete system with specified id.	Application_URL/services/api/v1/systems({systemId})
POST	Allows to configure equipment against a particular system	Application_URL/services/api/v1/systems(systemId)/topologyConfiguration
POST	Create connections for the equipment for a given system	Application_URL/services/api/v1/systems(systemId)/topologyConnections
PUT	Publish one or more system	Application_URL/services/api/v1/systems/publish
PUT	Creates a new revision of a system	Application_URL/services/api/v1/systems({systemId})/revise
PUT	System Publish	Application_URL/services/api/v1/systems({systemId})/publish
PUT	Change Lifecycle of the system	Application_URL/services/api/v1/systems({systemId})/lifecycle
GET	List of templates associated with specified system ID	Application_URL/services/api/v1/systems({systemId})/templates

Method	Actions	URI
PUT	Modify the list of templates associated with the specified system ID	Application_URL/services/api/v1/systems({systemId})/templates
GET	List of templates associated with specific system ID	Application_URL/services/api/v1/systems({systemId})/values
PUT	Modify the list of templates associated with specific system ID	Application_URL/services/api/v1/systems({systemId})/values
GET	Gets list of attached system ID	Application_URL/services/api/v1/systems({systemId})/components
PUT	Updates system components	Application_URL/services/api/v1/systems({systemId})/components
PATCH	Updates component structure of the system	Application_URL/services/api/v1/systems({systemId})/components
GET	Gets list of specified ID of system	Application_URL/services/api/v1/systems({systemId})/documents
POST	Updates system documents	Application_URL/services/api/v1/systems({systemId})/documents
GET	Get instructions assigned to the system	Application_URL/services/api/v1/systems({systemId})/instructions
PUT	Add or remove multiple instructions to or from multiple systems	Application_URL/services/api/v1/systems/instructions
GET	Get all locations	Application_URL/services/api/v1/systems({systemId})/locations
GET	Get all timeline for a system	Application_URL/services/api/v1/systems({systemId})/timeline
PUT	Add/remove/update the quantities of multiple parts assigned to a system	Application_URL/services/api/v1/systems/spareparts
GET	Retrieves all the spareparts assigned to the system	Application_URL/services/api/v1/systems({systemId})/spareparts

25.1 System Topology

System Topology enables you to create Industrial network topology, defining ports and interfaces and connections between them.

This can be compared to a laptop for example where we have an interface USB which have multiple ports and can be used to establish connection with other equipment such as a USB Pen Drive.

As a rule, for topologies:

- Every Equipment can have 1...n Interfaces.
- Every Interface can have 1...n Ports.
- Port of one Equipment will connect to a Port of another Equipment to establish connectivity.

Pre-requisites for creating a topology:

- To allow for creation of system topologies, you must create a model of type system and add communication systems as the subclass to be able to create topologies out of that system model.

Configure ports and interfaces of Equipment:

This API allows you to configure one or more equipment with ports and interfaces against a particular system.

- **HTTP Method:** POST
- **Example URL:** Application_URL/services/api/v1/systems(systemId)/topologyConfiguration
- **Authorization:** You must have the roles SYSTEM_DELETE or SYSTEM_EDIT assigned to your user ID.
- **Request Schema:**

Code Syntax

```
{
  "equipments":
  [{
    "id": {
      "type": "string",
      "maxLength": 32
    },
    "interfaces": [{
      "id": {
        "type": "string",
        "maxLength": 100
      },
      "ports": [{
        "id": {
          "type": "string",
          "maxLength": 100
        }
      ]
    }
  ]
}]
}
```

An example request for this schema can be:

Sample Code

```
{
  "equipments": [{
    "id": "ADC00D67ABC944F5AB764B9529045DCD",
    "interfaces": [{
      "id": "1",
      "ports": [{
        "id": "1"
      }],
      {
        "id": "2"
      },
      {
        "id": "3"
      }
    ]
  }, {
    "id": "2",
    "ports": [{
```

```

        "id": "1"
      },
      {
        "id": "2"
      },
      {
        "id": "3"
      }
    ]
  }
}, {
  "id": "0250E64BBB4A4051B08BE5FCBE636324",
  "interfaces": [{
    "id": "1",
    "ports": [{
      "id": "1"
    }, {
      "id": "2"
    }]
  }, {
    "id": "2",
    "ports": [{
      "id": "1"
    }, {
      "id": "2"
    }, {
      "id": "3"
    }, {
      "id": "4"
    }]
  }]
}, {
  "id": "734F1177CD3145F5AB3304DE08F31DEF",
  "interfaces": [{
    "id": "1",
    "ports": [{
      "id": "1"
    }, {
      "id": "2"
    }]
  }, {
    "id": "2",
    "ports": [{
      "id": "1"
    }, {
      "id": "2"
    }, {
      "id": "3"
    }, {
      "id": "4"
    }]
  }]
}
]
}

```

The above example allows us to add interfaces to 3 different equipment referenced by their equipment id and configure multiple interfaces along with their corresponding ports. The ports and interfaces have been given an id using numbers in the above example but any string can be used for the same.

Create connections among equipment using interfaces and ports:

This API allows you to create connections for the equipment that are configured ports and interfaces for a given system.

The request for the API is as follows:

- **HTTP Method:** POST
- **Example URL:** Application_URL/services/api/v1/systems(systemId)/topologyConfiguration
- **Authorization:** You must have the roles SYSTEM_DELETE or SYSTEM_EDIT assigned to your user ID.
- **Request Schema:**

Code Syntax

```
[{
  "sourceEquipment": {
    "type": "string",
    "maxLength": 32
  },
  "targetEquipment": {
    "type": "string",
    "maxLength": 32
  },
  "sourceInterfaceId": {
    "type": "string",
    "maxLength": 100
  },
  "targetInterfaceId": {
    "type": "string",
    "maxLength": 100
  },
  "sourcePortId": {
    "type": "string",
    "maxLength": 100
  },
  "targetPortId": {
    "type": "string",
    "maxLength": 100
  }
}]
```

An example request for this schema can be:

Sample Code

```
[{
  "sourceEquipment": "ADC00D67ABC944F5AB764B9529045DCD",
  "targetEquipment": "0250E64BBB4A4051B08BE5FCBE636324",
  "sourceInterfaceId": "1",
  "targetInterfaceId": "1",
  "sourcePortId": "1",
  "targetPortId": "2"
},
{
  "sourceEquipment": "ADC00D67ABC944F5AB764B9529045DCD",
  "targetEquipment": "734F1177CD3145F5AB3304DE08F31DEF",
  "sourceInterfaceId": "1",
  "targetInterfaceId": "1",
  "sourcePortId": "2",
  "targetPortId": "2"
}]
```

]


In this example we are taking 3 equipment and are creating connections amongst them via their configured interfaces and the ports available in those interfaces to create a topology. A port to port connection is created between the equipment from this API.

26 Templates

A model template is a format created by the manufacturer to maintain metadata related to an equipment.

It inherits a subclass template and has additional attribute groups and attributes. A template inherits the structure from a subclass template that is provided by the product and is based on ISO/DIS standards such as 14224.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles TEMPLATE_READ, TEMPLATE_EDIT, or TEMPLATE_DELETE assigned to your user ID.
- **API hub link:** [Template API](#) 

Methods

Group	Method	Actions	URI
Attribute	POST	Creates an attribute	Application_URL/ services/api/v1/attributes
	GET	Retrieves attribute list	Application_URL/ services/api/v1/attributes
	GET	Retrieves number of attributes	Application_URL/ services/api/v1/attributes/ \$count
	GET	Retrieves an attribute	Application_URL/ services/api/v1/attributes/ {attributeID}
	PUT	Updates an attribute	Application_URL/ services/api/v1/attributes/ {attributeID}
	DELETE	Deletes an attribute	Application_URL/ services/api/v1/attributes/ {attributeID}
	GET	Retrieves the equipments where this attribute has a value	Application_URL/ services/api/v1/attributes/ {attributeID}/equipment
	POST	Creates an attribute group	Application_URL/ services/api/v1/attribu- tegroups
	GET	Retrieves attribute groups	Application_URL/ services/api/v1/attribu- tegroups

Group	Method	Actions	URI
	GET	Retrieves the number of attribute groups	Application_URL/ services/api/v1/attributegroups/\$count
	GET	Retrieves an attribute group	Application_URL/ services/api/v1/attributegroups/{attributeGroupID}
	PUT	Updates an attribute group	Application_URL/ services/api/v1/attributegroups/{attributeGroupID}
	DELETE	Deletes an attribute group	Application_URL/ services/api/v1/attributegroups/{attributeGroupID}
Indicator	POST	Creates an indicator	Application_URL/ services/api/v1/indicators
	GET	Returns all indicators based on filter condition	Application_URL/ services/api/v1/indicators
	GET	Retrieves count of indicators	Application_URL/ services/api/v1/indicators/\$count
	GET	Retrieves an indicator	Application_URL/ services/api/v1/indicators/{indicatorID}
	PUT	Updates an indicator	Application_URL/ services/api/v1/indicators/{indicatorID}
	DELETE	Deletes an indicator	Application_URL/ services/api/v1/indicators/{indicatorID}
	POST	Creates an indicator group	Application_URL/ services/api/v1/indicatorgroups
	GET	Retrieves the indicator groups	Application_URL/ services/api/v1/indicatorgroups
	GET	Retrieves the count of indicator groups	Application_URL/ services/api/v1/indicatorgroups/\$count
	GET	Retrieves an indicator group	Application_URL/ services/api/v1/indicatorgroups/{indicatorGroupID}
	PUT	Updates an indicator group	Application_URL/ services/api/v1/indicatorgroups/{indicatorGroupID}

Group	Method	Actions	URI
	DELETE	Deletes an indicator group	Application_URL/ services/api/v1/indica- torgroups/{indicatorGrou- pID}
CodeList	POST	Creates a codelist	Application_URL/ services/api/v1/codelists
	GET	Retrieves codelists	Application_URL/ services/api/v1/codelists
	GET	Retrieves count of codelists	Application_URL/ services/api/v1/codelists/ \$count
	GET	Retrieves a codelist	Application_URL/ services/api/v1/codelists/ {codelistID}
	PUT	Updates a codelist	Application_URL/ services/api/v1/codelists/ {codelistID}
	DELETE	Deletes a codelist	Application_URL/ services/api/v1/codelists/ {codelistID}
Template	POST	Creates a template	Application_URL/ services/api/v1/templates
	GET	Retrieves templates	Application_URL/ services/api/v1/templates
	DELETE	Deletes the list of templates	Application_URL/ services/api/v1/templates
	GET	Retrieves the count of tem- plates	Application_URL/ services/api/v1/templates/ \$count
	GET	Retrieves a template using internalId	Application_URL/ services/api/v1/templates/ internalid/{internalId}
	GET	Retrieves a template	Application_URL/ services/api/v1/templates/ {templateID}
	PUT	Updates a template	Application_URL/ services/api/v1/templates/ {templateID}
	DELETE	Deletes a template	Application_URL/ services/api/v1/templates/ {templateID}
	GET	Retrieves template types	Application_URL/ services/api/v1/template- types

Group	Method	Actions	URI
Industry Standards	GET	Retrieves industry standards associated with the specified template	Application_URL/ services/api/v1/templates/ {templateID}/industrystan- dards
	PUT	Updates industry standards	Application_URL/ services/api/v1/templates/ {templateID}/industrystan- dards
	DELETE	Deletes industry standards	Application_URL/ services/api/v1/templates/ {templateID}/industrystan- dards
	GET	Retrieves industry standard codes	Application_URL/ services/api/v1/industrys- tandards
Alert Type	GET	Returns alert types	Application_URL/ services/api/v1/alerttypes
	PUT	Updates an alert type	Application_URL/ services/api/v1/alerttypes
	POST	Create alert types	Application_URL/ services/api/v1/alerttypes/ mass
	DELETE	Deletes an alert type	Application_URL/ services/api/v1/alerttypes/ {alerttypeID}
	POST	Create alert type groups	Application_URL/ services/api/v1/alertty- pegroups
	GET	Retrieve alert type groups	Application_URL/ services/api/v1/alertty- pegroups
	PUT	Updates an alert type group	Application_URL/ services/api/v1/alertty- pegroups
	DELETE	Deletes an alert type group	Application_URL/ services/api/v1/alertty- pegroups/{alerttypegroupID}
Nested Structures	POST	Creates a nested structure	Application_URL/ services/api/v1/nested/ structures
	GET	Retrieves a nested structure by Internal Id	Application_URL/ services/api/v1/nested/ structures

Group	Method	Actions	URI
	PUT	Updates a nested structure	Application_URL/ services/api/v1/nested/ structures/{nestedStructur- eID}
	GET	Retrieves a nested structure	Application_URL/ services/api/v1/nested/ structures/{nestedStructur- eID}
	DELETE	Deletes a nested structure	Application_URL/ services/api/v1/nested/ structures/{nestedStructur- eID}

26.1 Additional Information

Attribute Request Payload Parameters

Field (Mandatory or Optional)	Data Type	Description
<internalId>(Mandatory)	String	a mandatory field that defines a unique ID for the attribute.
<div style="background-color: #f0f0f0; padding: 5px; border: 1px solid #ccc;"> <p>i Note</p> <p><name> has been deprecated.</p> </div>		
<Id>	String	a unique identifier generated by the system for an attribute.
<Description>	String	a description for an attribute. It has the following subfields: <ul style="list-style-type: none"> • <Id>: unique identifier that is system generated for an attribute • <short>: defines a short description for an attribute • <long>: defines a long description for an attribute

Field (Mandatory or Optional)	Data Type	Description
<code><dataType></code>	String	<p>Defines the kind of data types that can be assigned for an attribute.</p> <p>List of supported data types are:</p> <ul style="list-style-type: none"> • boolean • string • numeric • date • enum • picture <p>For more information on using the data types and defining additional properties for these data types, see Defining Additional Properties for a Data Type</p>
dimension1	String	<p>a field that defines a unit of measurement for an numeric attribute</p> <p>For more information on supported dimensions, refer the list provided in dimension2 description.</p>
dimension2	String	<p>a field that defines additional unit of measurement for an numeric attribute.</p> <p>List of supported dimensions are:</p> <ul style="list-style-type: none"> • weight • length • time • force • mass • pressure • current • energy • power • accleration • electric current • temperature
<code><enumValues></code>	array	<p>Allows you to specify an array of values which serves as an enumeration for enum attribute.</p>

Field (Mandatory or Optional)	Data Type	Description
unitOfMeasure1	array	Specifies the allowed values for dimension1
unitOfMeasure2	array	Specifies the allowed values for dimension2
<isMultivalue>	boolean	Specifies if the enum attribute can support multiple values.
<isLanguage>	boolean	Specifies if the value supported by a string attribute is language-dependent or not.
<isRange>	boolean	Specifies if the numeric attribute can support a range.
<industryStandards>	array	<p>Specifies the supported industry standards at the attribute level. Possible values:</p> <ul style="list-style-type: none"> • ISO 15926 • IEC 60050 • ISO 14224 • IO-LINK • MIMOSA <p>It has the following sub fields:</p> <ul style="list-style-type: none"> • industryStandard: specifies the name of the industry standard. • standard ID: a unique identifier corresponding to an industry standard specified by the user.
<hasNormalValue>	boolean	Specifies if the numeric attribute can support a normal value. This flag can be used only if <isRange> is set to true.
optionCode	Integer	Specifies the priority - Optional, Mandatory or Recommended

Attribute Group Fields

Field (Mandatory or Optional)	Data Type	Description
<code><internalId></code> (Mandatory)	String	a mandatory field that defines a unique ID for the attribute group.
<div style="background-color: #f0f0f0; padding: 5px;"> <p>i Note <code><name></code> has been deprecated.</p> </div>		
<code><Id></code>	String	a unique identifier for an attribute group that is generated by the system.
<code><Description></code>	String	<p>a description for an attribute group. It has the following subfields:</p> <p>Id: unique identifier for an attribute group</p> <p>short: defines a short description for an attribute group</p> <p>long: defines a long description for an attribute group</p>
<code><attributes></code>	array	An array of IDs that uniquely identifies each attribute.

Template Attributes

Field (Mandatory or Optional)	Data Type	Description
<code><internalId></code> (Mandatory)	String	a mandatory field and defines a unique ID for the template.
<div style="background-color: #f0f0f0; padding: 5px;"> <p>i Note <code><name></code> has been deprecated.</p> </div>		
<code><Id (Optional)></code>	String	a system generated unique identifier for a template.
<code><description (Optional)></code>	String	<p>Defines a description for a template. It has the following subfields:</p> <ul style="list-style-type: none"> <code><Id></code>: unique identifier for the template <code><short></code>: defines a short description for a template <code><long></code>: defines a long description for a template.
<code><parentId (Optional)></code>	String	a unique identifier of a parent template.

Field (Mandatory or Optional)	Data Type	Description
<code><isAbstract (Optional)></code>	boolean	a flag that identifies if a template is abstract or non-abstract.
<code><type (Mandatory)></code>	String	<p>a mandatory field that defines the type of a template. It can take any of the following three values:</p> <ul style="list-style-type: none"> • class (1): A class template is a base template. • subclass (2): A subclass template inherits from a class template. • model template (3): A model template inherits either from a subclass template or another model template. • equipment template (4): An equipment template is independent of class or subclass. It can have separate attributes of its own. • location template (5): A location template contains location-specific attributes and attribute groups related to a location. • spare part template(6): A spare part template contains spare part-specific attributes and attribute groups. • system template(7): A system template contains system-specific attributes and attribute groups. • functional location template(8): A functional location template contains functional location-specific attributes and attribute groups.
<code><attributeGroups (Optional)></code>	array	An array of IDs that uniquely identifies each attribute group.
<code><changedOn (Optional)></code>	String	Specifies the date when the template was last updated.

Indicators

Field (Mandatory or Optional)	Data Type	Description
<code><internalId></code> (Mandatory)	String	a mandatory field and defines a unique ID for the indicator.
<div style="background-color: #f0f0f0; padding: 5px; border: 1px solid #ccc;"> <p>i Note <code><name></code> has been deprecated.</p> </div>		
<code><description></code>	String	Defines a description for a template. It has the following subfields: <ul style="list-style-type: none"> <code><short></code> (Mandatory): defines a short description for an indicator <code><long></code>: defines a long description for an indicator
<code><descriptions></code>	Sstring	Specifies the list of descriptions of the object in various languages. This takes priority over <code><description></code> field <ul style="list-style-type: none"> <code><language></code>: Defines a language for the description object. <code><short></code> (Mandatory): defines a short description for an indicator <code><long></code>: defines a long description for an indicator
<code><dataType></code> (Mandatory)	String	Defines the kind of data types that can be assigned for an indicator.
<code><aggregationConcept></code> (Mandatory)	String	The aggregation concept used to show the indicator values.
<code><indicatorGroup></code>	String	
<code><isInstantiationRequired></code>	array	An array of IDs that uniquely identifies each attribute group.
<code><scale></code>	Integer	Reserves the number of significant places after decimal.
<code><precision></code>	Integer	Specifies the maximum number of digits numeric flexible value can have including decimal places.
<code><hasThreshold></code>	Boolean	To be set if user wants to specify threshold as Min/Max/Normal. Should be used along with <code>isRange</code> .
<code><dimension1></code>	String	Dimension 1 of the indicator in case of numeric data type only.

Field (Mandatory or Optional)	Data Type	Description
<expectedBehaviour>	String	The expected behaviour of the indicator.

Indicator Groups

Field (Mandatory or Optional)	Data Type	Description
<description (Optional)>	String	Specifies the list of descriptions of the object in various languages. This takes priority over <description> field <ul style="list-style-type: none"> • <short>: defines a short description for a template • <long>: defines a long description for a template. • <language>: unique identifier for the template
<internalId>	String	A mandatory field that defines a unique internal ID for the indicator group.
<indicators>	String	List of IDs of attributes which are to be added to the indicator group.

Code Lists

Field (Mandatory or Optional)	Data Type	Description
<internalId>(Mandatory)	String	A mandatory field that defines a unique internal ID for a codelist.
<dataType>	String	Defines the kind of data types that can be assigned for a codelist.
<description (Optional)>	String	Defines a description for a template. It has the following subfields: <ul style="list-style-type: none"> • <language>: unique identifier for the template • <short>: defines a short description for a template • <long>: defines a long description for a template.
<items>	array	Specifies the list of descriptions of the codes.
<industryStandards>	String	The list of industry standards the code list is complaint to.

27 Unit of Measure Conversion

Unit of Measure (UoM) conversion deals with various conversions across units within a dimension. The unit system defines a source unit that must be converted to a target unit. For example, if the unit system is metric and dimension is length, any source unit (miles) must be converted to target unit, (kms).

By default, unit systems, metric (selected) and imperial are available. You can personalize the unit system using the Fiori launchpad.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **API hub link:** [Unit of Measures API](#)

Methods

Group	Method	Actions	URI
Unit Systems Associations	POST	Converts target unit values	Application_URL/ services/api/v1/uom/convert
List and Header	GET	Retrieves list of dimensions and units associated unit systems based on the boolean parameter	Application_URL/ services/api/v1/uom/dimensions
	GET	Retrieves list of dimensions and units associated unit systems based on the dimension ID parameter	Application_URL/ services/api/v1/uom/dimensions/{dimensionId}
	POST	Identifies possible target units	Application_URL/ services/api/v1/uom/targetunits
	GET	Retrieves list of unit systems	Application_URL/ services/api/v1/uom/ unitsystems
	GET	Retrieves list of unit systems along with unit mappings	Application_URL/ services/api/v1/uom/ unitsystems/{unitSystem}
	POST	Retrieves details of the unit system Retrieves details of the unit system	Application_URL/ services/api/v1/uom/ unitsystems/{unitSystem}

28 User Authorization

You use this feature to define a more granular access to asset modeling business objects for users within your organization.

i Note


User Authorization is an optional feature, which can be used in addition to the existing [Network Authorizations](#).

All the users assigned to a node in the organizational hierarchy have access to:

- The business objects and groups assigned to the relevant node in the hierarchy.
- The business objects and groups assigned to the subnodes of a relevant parent node in the organizational hierarchy.

This API allows you to create, update, retrieve, delete organizational hierarchy.

Additional Information

- **Base URI:** `Application_URL/services/api/v1`
- **Permissions:** You must have the roles `USER_AUTH_READ`, `USER_AUTH_DELETE`, or `USER_AUTH_EDIT` assigned to your user ID.
- **API hub link:** [User Authorizations API](#) 

Methods


Method	Actions	URI
POST	Creates an organization unit	<code>Application_URL/services/api/v1/user-auth/organization/units</code>
DELETE	Deletes an organization unit	<code>Application_URL/services/api/v1/user-auth/organization/units({groupId})</code>
GET	Returns organization unit header info	<code>Application_URL/services/api/v1/user-auth/organization/units({groupId})/header</code>
PUT	Updates organization unit header info	<code>Application_URL/services/api/v1/user-auth/organization/units({groupId})/header</code>
PUT	Add, remove, or update users in an organization unit	<code>Application_URL/services/api/v1/user-auth/organization/units({groupId})/members</code>
PUT	Add or remove business objects from an organizational unit	<code>Application_URL/services/api/v1/user-auth/organization/units({groupId})/objects</code>

Method	Actions	URI
GET	Retreives organizational unit structure	Application_URL/services/api/v1/user-auth/organization/units/structure

29 Work Orders

A work order facilitates maintenance of equipment and other related processes.

Additional Information

- **Base URI:** Application_URL/services/api/v1
- **Permissions:** You must have the roles EQUIPMENT_EDIT or EQUIPMENT_DELETE for equipment and FUNCTIONAL_LOCATION_EDIT or FUNCTIONAL_LOCATION_DELETE for locations assigned to your user ID.
- **API hub link:** [Work Orders API](#) 

Methods

Method	Actions	URI
POST (Deprecated from 2011 release)	Creates a work order	Application_URL/services/api/v1/workorder
PUT (Deprecated from 2011 release)	Updates a work order	Application_URL/services/api/v1/workorder
GET	Retrieves work orders	Application_URL/services/api/v1/workorders
GET	Retrieves the count of work orders	Application_URL/services/api/v1/workorders/\$count

30 Searching Objects in SAP Asset Intelligence Network

You use this API to search objects in SAP Asset Intelligence Network such as models, equipment, subclass, and templates. Since the search is fault-tolerant, you can search using terms that might not be a perfect match with the object.

You can also narrow down your search by using the following:

Path Parameter	Objects
MOD	Models
EQU	Equipment
TEM	Model templates, equipment templates and location templates
CAT	Subclass
ALL	Models, Equipment, Model Templates, Equipment Templates, Location Templates and Subclass

Request

- *HTTP Method:* GET
- *URL:* `Application_URL/services/api/v1/searchObject/<path parameter>?searchText=<Provide space-separated search terms which can be Manufacturer name, Manufacturer ID, Industry group name, Object name, description, Class name, Subclass name, Template name, Operator name, Source name>`
- *Query Parameters:* For details about the query parameter, see

Response

Based on the query parameters, the system returns the objects you searched for.

<https://help.sap.com/http.svc/rc/b7b5a3bbe4564f3b8b06566da64a4534/1702/en-US/fuzzysearch.html>

For more information, see <https://help.sap.com/http.svc/rc/b7b5a3bbe4564f3b8b06566da64a4534/1702/en-US/fuzzysearch.html>.

31 OData Services for SAP Analytics Cloud

OData Services allows you to import and use data from multiple business objects of SAP Asset Intelligence Network to feed analytics dashboards.

Objects

You can import and use header data or last indicator values from the following business objects of SAP Asset Intelligence Network using different APIs:

- Equipment
- Indicators
- Models
- Locations
- Spare parts
- Notifications
- Work orders

Related Information

[Configuring SAP Analytics Cloud and Offline Analytics Dashboards](#)

31.1 Reading Data of Equipment

This API allows you to read data of equipment.

Request

- Format: *XML*
- HTTP Method: *GET*
- URI: `https://<launchpad-url>/ain/ac.odata.svc/api/v1/Equipment`

Response

The system returns the header data of equipment.

31.2 Reading Data of Indicators

This API allows you to read data of indicators.

Request

- Format: *XML*
- HTTP Method: *GET*
- URI: `https://<launchpad-url>/ain/ac.odata.svc/api/v1/EquipmentIndicatorValues`

Response

The system returns the last indicator values of equipment.

31.3 Reading Data of Models

This API allows you to read data of models.

Request

- Format: *XML*
- HTTP Method: *GET*
- URI: `https://<launchpad-url>/ain/ac.odata.svc/api/v1/Models`

Response

The system returns the header data of models.

31.4 Reading Data of Locations

This API allows you to read data of locations.

Request

- Format: *XML*
- HTTP Method: *GET*
- URI: `https://<launchpad-url>/ain/ac.odata.svc/api/v1/Locations`

Response

The system returns the header data of locations.

31.5 Reading Data of Spare Parts

This API allows you to read data of spare parts.

Request

- Format: *XML*
- HTTP Method: *GET*
- URI: `https://<launchpad-url>/ain/ac.odata.svc/api/v1/SpareParts`

Response

The system returns the header data of spare parts.

31.6 Reading Data of Notifications

This API allows you to read data of notifications.

Request

- Format: *XML*
- HTTP Method: *GET*
- URI: `https://<launchpad-url>/ain/ac.odata.svc/api/v1/Notifications`

Response

The system returns the header data of notifications.

31.7 Reading Data of Work Orders

This API allows you to read data of work orders.

Request

- Format: *XML*
- HTTP Method: *GET*
- URI: `https://<launchpad-url>/ain/ac.odata.svc/api/v1/WorkOrders`

Response

The system returns the header data of work orders.

32 Asynchronous Operations

This API allows you to monitor the Reliability Centered Maintenance (RCM) reassessment operations that are asynchronous.

Additional Information

- **Base URI:** `Application_URL/aspm/asyncops/api/v1/objects/{objectId}/operations`
- **Permissions:** You must have the roles `RISK_ASSESSMENT_READ`, `RISK_ASSESSMENT_EDIT`, or `RISK_ASSESSMENT_DELETE` assigned to your user ID.
- **HTML link:** <https://help.sap.com/doc/971fafd93d864a2098a63afd6b5d2f5d/2111/en-US/AsyncOps.html>



Method	Actions	Application URL
GET	Retrieves all operation for a single object.	<code>Application_URL/aspm/asyncops/api/v1/objects/{objectId}/operations</code>
GET	Retrieves all operations for an object.	<code>Application_URL/aspm/asyncops/api/v1/objects/{object-Type}/{objectId}/operations</code>
GET	Get all operations.	<code>Application_URL/aspm/asyncops/api/v1/operations</code>
POST	Initiate new operation.	<code>Application_URL/aspm/asyncops/api/v1/operations</code>
GET	Get operation details.	<code>Application_URL/aspm/asyncops/api/v1/operations/{operationId}</code>
PUT	Update operation status.	<code>Application_URL/aspm/asyncops/api/v1/operations/{operationId}/status</code>
PUT	Update operation tasks.	<code>Application_URL/aspm/asyncops/api/v1/operations/{operationId}/tasks</code>
PUT	Update operation task status.	<code>Application_URL/aspm/asyncops/api/v1/operations/{operationId}/tasks/{taskId}</code>
GET	Get all operations started by user.	<code>Application_URL/aspm/asyncops/api/v1/users/self/operations</code>

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 a 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.

© 2022 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.