How-to Guide | PUBLIC

2026-10-07

Custom Code Migration Guide for SAP S/4HANA 2025

Feature Package Stack 02

© 2026 SAP SE or an SAP affiliate company. All rights reserved.

Content

1 Custom Code Migration Guide for SAP S/4HANA 2025. 5

1.1 System Conversion. 5

1.2 Process. 6

2 Custom Code Analysis Before System Conversion. 9

2.1 Using SAP S/4HANA 2025 as Central Check System. 10

Getting Started. 11

Preparing the Custom Code Analysis. 18

Custom Code Analysis with SAP Fiori App (Recommended). 21

2.2 Using SAP NetWeaver Application Server for ABAP 7.52 as Central Check System. 28

Getting Started. 29

Preparing the Custom Code Analysis. 34

Custom Code Analysis in SAP GUI. 37

2.3 Using SAP BTP ABAP environment as Central Check System (Recommended Solution by SAP) 42

Automated Setup (Recommended). 42

Manual Setup. 43

Custom Code Analysis with SAP Fiori App (Recommended). 55

3 Custom Code Adaptation After System Conversion. 64

3.1 Running Transactions SPDD, SPAU, and SPAU_ENH. 64

3.2 Running Local SAP S/4HANA Checks. 65

Importing the Simplification Database. 65

Configuring Local ATC Run Series. 65

Scheduling Local ATC Run Series. 67

3.3 Adapting Custom Code in ABAP Development Tools for Eclipse. 68

Displaying Active Results. 68

Optional: Changing the Contact Person of ATC Findings. 69

Fixing ATC Findings. 70

Rechecking the Active Result. 77

3.4 Rerunning Local SAP S/4HANA Checks. 77

4 What's Next?. 78

5 Glossary. 80

Document History

Version	Date	Description
SAP S/4HANA 2023 FPS00	2023-10-11	Updated chapter Maintaining Communication Arrangements [page 52]. Updated chapter Changing the Scope of Your Custom Code Migration Project [page 25].
SAP S/4HANA 2022 FPS00	2022-10-12	Added chapter Using Exemptions [page 76].
SAP S/4HANA 2021 FPS01	2022-02-23	Added a Document History [page 3]. Changes in chapter Using SAP BTP ABAP environment as Central Check System (Recommended Solution by SAP) [page 42]: • Added subchapter Automated Setup (Recommended) [page 42].

Version	Date	Description
SAP S/4HANA 2021 FPS00	2021-10-13	Changes in chapter Using SAP S/4HANA 2025 as Central Check System [page 10]: • Updated RFC user authorizations in subchapter Configuring the User [page 11]. • Added subchapter Adding Usage Data Stored in SUSG [page 23]. • Added subchapter Adding Usage Data Stored in SAP Solution Manager [page 24]. Changes in chapter Using SAP NetWeaver Application Server for ABAP 7.52 as Central Check System [page 28]: • Updated RFC user authorizations in subchapter Configuring the User [page 29]. Changes in chapter Using SAP BTP ABAP environment as Central Check System (Recommended Solution by SAP) [page 42]: • Updated RFC user authorizations in subchapter Configuring the User [page 44]. • Added subchapter Adding Usage Data Stored in SUSG [page 57]. • Added subchapter Adding Usage Data Stored in SAP Solution Manager [page 58].

1 Custom Code Migration Guide for SAP S/4HANA 2025

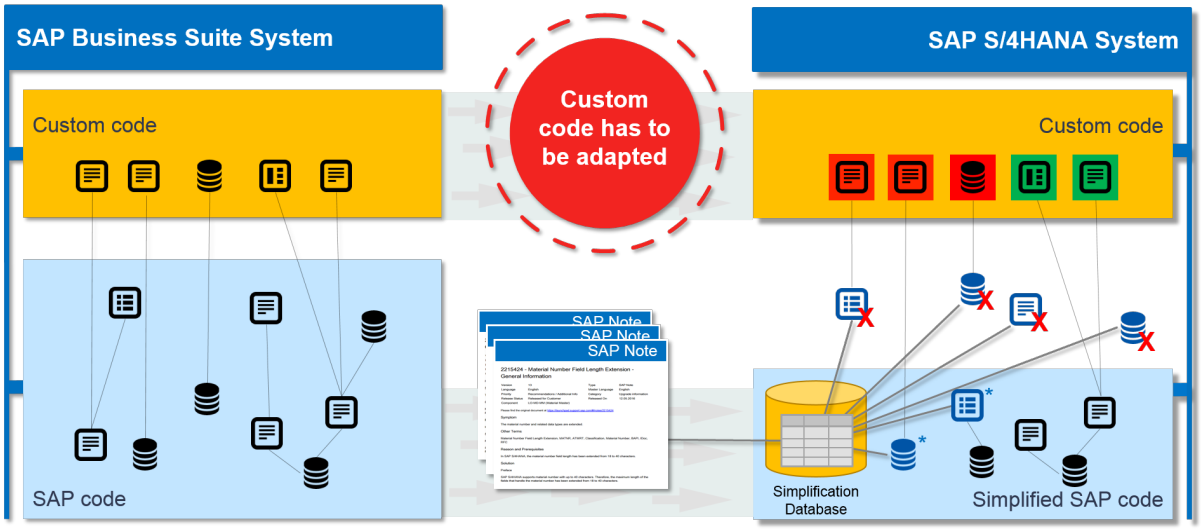
This guide focuses on the custom code related process from checking custom code remotely, analyzing the findings, and finally adapting the custom code to SAP S/4HANA 2025.

Read this guide to get instructions on how to adapt your custom code to SAP S/4HANA 2025.

> **Note**
>
> Starting a migration project also provides the opportunity to think about a general re-organization of your custom code. For this purpose, we recommend you make yourself familiar with the new extensibility capabilities and the new technologies in SAP S/4HANA. You'll find a comprehensive overview here: Custom Extensions in SAP S/4HANA Implementations

1.1 System Conversion

Custom Code Migration is part of the system conversion process from the classic SAP Business Suite running on any database to the SAP S/4HANA system. In the context of this system conversion, custom ABAP source code needs to be adapted, since a lot of SAP code within SAP S/4HANA was simplified and in some cases changed in a non-compatible way.



Some of your custom code objects are not valid anymore and either do not perform as expected or produce syntax errors or dumps (red objects in the picture). You almost certainly have other objects that do perform as expected and do not need to be changed (green objects in the picture).

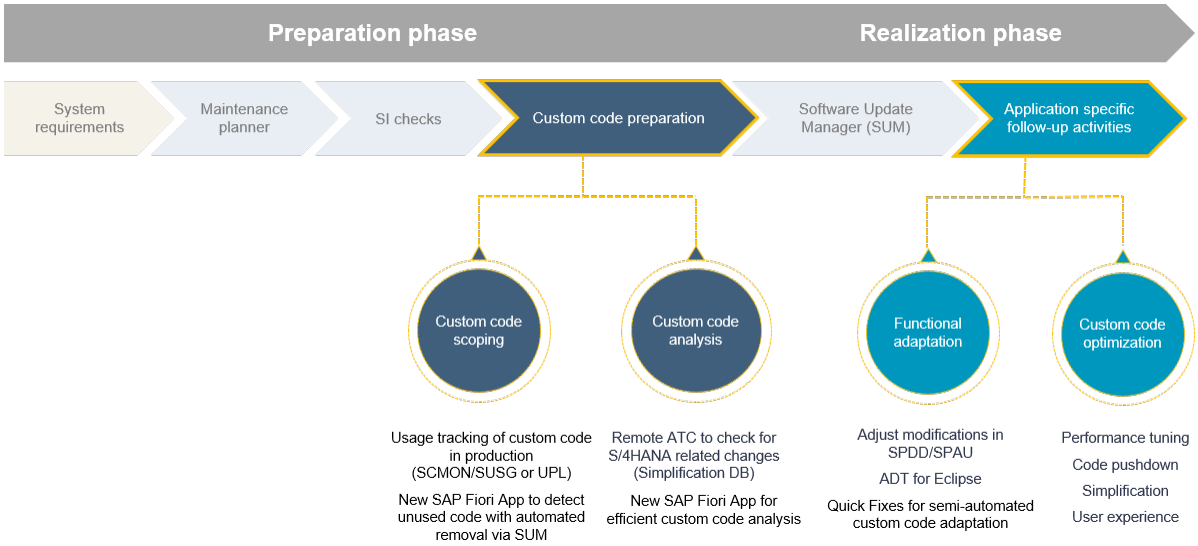
SAP provides tools, based on the Simplification Database, that detect any custom code that needs to be adapted to SAP S/4HANA. The Simplification Database is a database table in the SAP S/4HANA system that

contains all Simplification Items that refer to SAP objects simplified in SAP S/4HANA. Each simplification item describes changed or removed SAP objects and refers to a dedicated SAP Note that describes the impact of the change and how the related custom code can be adapted.

1.2 Process

The custom code migration process for the system conversion to SAP S/4HANA describes the tools and necessary procedures to migrate your custom code. The process consists of the preparatory analysis (custom code scoping and custom code analysis) and the adaptation of the custom code (custom code adaptation) after the technical system conversion.

The following figure provides you with an overview of the phases, the activities, and the tools involved in this process.



Overview of the conversion process that is divided into the preparation phase and the realization phase

For more information, see:

SAP S/4HANA System Conversion – Custom code adaptation process

Conversion Guide for SAP S/4HANA

Custom Code Scoping

Custom code scoping enables you to determine the ABAP custom code that is frequently used and needs to be taken over to SAP S/4HANA. Unused custom code can be deleted using deletion transport requests during the system conversion. This means, scoping enables you to reduce the number of custom code to be migrated and to minimize the efforts of the custom code adaptation.

> **Note**
>
> Scoping is only supported using the `Custom Code Migration` app.

→ Recommendation

SAP recommends to scope your custom code in order to delete unused custom code and to minimize your efforts.

If you want to benefit from scoping, you need to monitor your productive system landscape in advance. To do this, SAP recommends

- Using the ABAP Call Monitor (transaction `SCMON`) and
- Transaction `SUSG` to collect usage data in production system for at least one year.

For more information, see

- ABAP Call Monitor (SCMON) – Analyze usage of your code
- Aggregate usage data in your production system with SUSG transaction
- Usage Data Collection

Custom Code Analysis

Custom code analysis enables you to identify the custom code that must be adapted in the context of the system conversion to SAP S/4HANA.

You have the following possibilities to run the custom code analysis:

- Using the `Custom Code Migration` app to
  - Run the SAP S/4HANA checks to analyze which custom code needs to be adapted.
  - Get statistical information about the findings in your development objects that need to be adapted.
- Using the classic **SAP GUI** to run the SAP S/4HANA checks based on the ABAP Test Cockpit (ATC).

Both options provide the relevant checks in a central check system. The central check system can remotely execute these checks on any ABAP system (`SAP_BASIS 7.00` or higher) where the custom code to be analyzed is stored.

Custom Code Adaptation (Functional Adaptation)

After the system conversion, you need to perform the functional adaptations in your custom code.

In the system conversion, you need to adapt any modifications related to ABAP Dictionary objects using transaction `SPDD`.

After you performed the system conversion to SAP S/4HANA with Software Update Manager (SUM), you need to adapt any modifications related to repository objects using transactions `SPAU` and `SPAU_ENH`.

After this, we recommend that you run ABAP Test Cockpit (ATC) with SAP S/4HANA checks in ABAP development tools for Eclipse. The result is a list of ATC findings that relate to SAP S/4HANA simplifications and refer to SAP Notes that describe how to solve the issues. Based on these ATC findings, you can start adapting your custom code using quick fixes in a semi-automated way.

→ Recommendation

SAP recommends you to run ABAP Test Cockpit (ATC) with SAP S/4HANA checks in *ABAP development tools for Eclipse*. Here, the check result is provided as a list containing the ATC findings that relate to SAP S/4HANA simplifications and refer to the SAP Notes which describe how to solve the issues. Based on these ATC findings and specific quick fixes, you can start adapting your custom code in a semi-automated way.

> **Note**
>
> You also need to adapt some of your custom code if you upgrade from a lower release to SAP S/4HANA 2025.

Performance Tuning

Finally, you need to check which business processes, for example performance of critical database queries, need to be optimized in SAP HANA. To do this, you need to check which SQL statements can be optimized.

The relevant performance data for all SQL statements executed in your productive system is displayed in the *SQL Monitor*. There, you can find the most expensive business processes and the most frequently used SQL statements. You can open the *SQL Monitor* via transaction `SQLM`.

Custom Code Migration Guide for SAP S/4HANA 2025

2 Custom Code Analysis Before System Conversion

Overview

Some options of the custom code analysis depend on whether your central check system is an on-premise system or based on SAP BTP ABAP environment.

Options

	SAP Fiori Custom Code Migration App	Remote ATC with SAP GUI	SAP Fiori Custom Code Migration App in SAP BTP
Required system	Central ATC check system **(SAP S/4HANA ≥1809)**	Central ATC check system **(SAP_BASIS 7.52)**	SAP BTP, ABAP Environment
System location	Customer landscape	Customer landscape	SAP BTP
Remote connectivity	Through RFC	Through RFC	Through RFC and Cloud Connector
Analyze SAP S/4HANA findings	✓	✓	✓
Simplification information in ATC result	✓	✓	✓
Filter results by scope and Quick Fix availability	✓	✗	✓
Define custom code migration scope based on usage data	✓	✗	✓
Remove unused code during system conversion via SUM	✓	✗	✓

Options to Run the Custom Code Analysis

Follow **one** of the subsequent options to run the custom code analysis:

1. If your central check system is based on **SAP S/4HANA 1809 or higher**, you can use the `Custom Code Migration` app to define the scope of your custom code that needs be migrated to SAP S/4HANA and to analyze custom code for SAP S/4HANA custom code checks. This SAP Fiori app acts as a central check system on the latest version of SAP S/4HANA. This ensures that you run the latest version of the SAP S/4HANA custom code checks.

   > → Tip
   >
   > If you choose this option, see Using SAP S/4HANA 2025 as Central Check System [page 10] for more information on how to run custom code scoping and custom code analysis.

2. If your central check system is based on **SAP NetWeaver Application Server for ABAP 7.52**, you can use an ABAP Test Cockpit (ATC) central check system instead. The latter analyzes custom code for SAP S/4HANA custom code checks.

With this approach you can neither scope your custom code based on usage data nor filter the analysis results by scope or quick fix availability.

> → Tip
>
> If you choose this option, see chapter Using SAP NetWeaver Application Server for ABAP 7.52 as Central Check System [page 28] for more information how to run the custom code analysis.

3. If your central check system is based on SAP BTP ABAP environment, you can use the `Custom Code Migration` app to define the scope of your custom code that needs to be migrated to SAP S/4HANA and to analyze custom code for SAP S/4HANA custom code checks. The SAP BTP ABAP environment will analyze your custom code in your on-premise system remotely through Cloud Connector. The SAP BTP ABAP environment is updated four times a year and contains the latest features of the `Custom Code Migration` app.

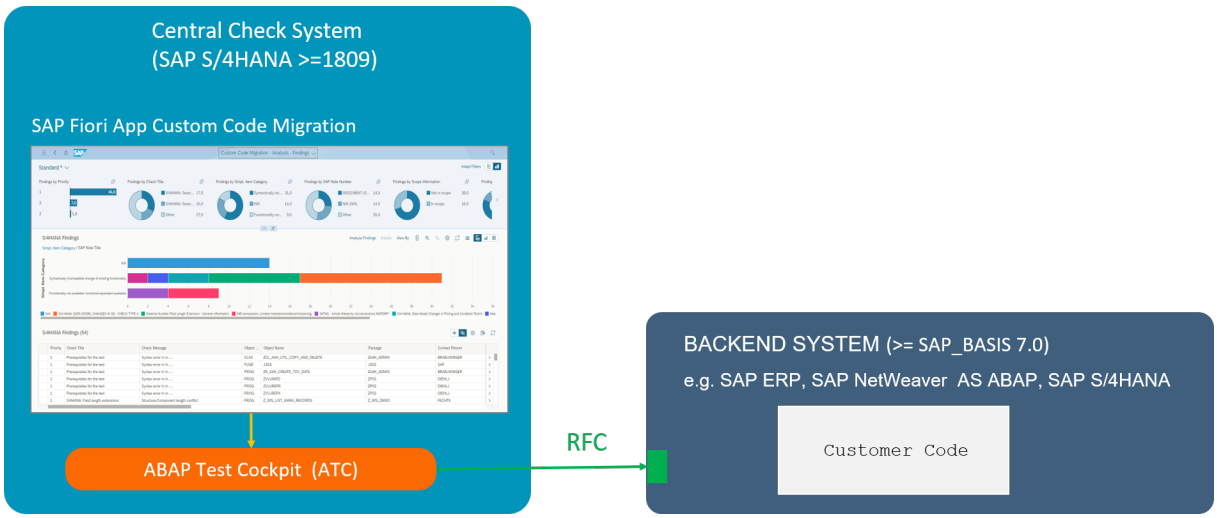
> → Tip
>
> If you choose this option, see Using SAP BTP ABAP environment as Central Check System (Recommended Solution by SAP) [page 42] for more information how to run the custom code scoping and custom code analysis.

2.1 Using SAP S/4HANA 2025 as Central Check System

If your central check system is based on **SAP S/4HANA 2025**, you use the `Custom Code Migration` app to scope and analyze custom code in your on-premise SAP system for SAP S/4HANA custom code checks. This ensures that you run the analysis based on the latest ATC check versions in the context of ABAP platform.

The `Custom Code Migration` app enables you to scope and analyze custom code that needs to be migrated from an SAP Business Suite system to SAP S/4HANA 2025.

To evaluate the development objects to be adapted, this SAP Fiori app performs the SAP S/4HANA custom code checks.



Setup of the remote custom code adaption of the check results investigated using the Custom Code Migration app

> **Note**
>
> If you have already set up an SAP S/4HANA Sandbox system of the latest release (2022), this system can also be used for the custom code analysis.
>
> If you have already set up an SAP S/4HANA Foundation system of the latest release (2022), this system will be technically sufficient to run the custom code migration app. For more information about SAP S/4HANA Foundation, see SAP Note 3015497.

2.1.1 Getting Started

2.1.1.1 System Requirements

To perform the custom code analysis with the `Custom Code Migration` app, you must set up an SAP S/4HANA 2023 system.

Moreover, you have to set up the SAP Fiori front-end server on the central check system. For more information, see Embedded Deployment of SAP Fiori Front-End Server.

SAP System	Software Component
Central Check System	`SAP_BASIS 7.56`
Checked System	SAP Business Suite system based on `SAP_BASIS 7.00` or higher

2.1.1.2 Configuring the User

You can either apply SAP Note 3548772 in the **checked system** to get all relevant authorizations for your RFC user or, alternatively, check the table below.

Authorization Object Value	Parameter	Value
S_RFC	ACTVT	16 (Execute)
	RFC_TYPE	FUGR

Authorization Object Value	Parameter	Value
	RFC_NAME	SCA_REMOTE_DATA_ACCESS
		SABP_COMP_PROCS_E
		SYCM_APS_REMOTE
		SYST
		S_CODE_INSPECTOR_TESTS
S_RFC	ACTVT	16 (Execute)
	RFC_TYPE	FUNC
	RFC_NAME	FUNCTION_EXISTS
		REPOSITORY_ENVIRONMENT_ALL
		RFC_GET_NAMETAB
		SVRS_GET_VERSION_DIRECTORY_46
		RFCPING
		SLINRMT_RUN
		TRINT_PROGRESS_INDICATOR
		TRINT_TP_UPDATE_TPSTAT
		SLDAG_GET_SYSTEM_NAME
		CVA_BSP_RUN
		IDOCTYPE_COLLECT_ATC_INFOS
		IQAPI_QUERY_CONTENT
		IQAPI_INFOSET_CONTENT
		IQAPI_CHECK_INFOSET
		FM_CCM_DYNPRO_READ
		SCI_GET_DDLS
S_DEVELOP	ACTVT	03 (Display)
	DEVCLASS	*
	OBJNAME	*
	OBJTYPE	*
	P_GROUP	*

Authorization Object Value	Parameter	Value
S_SYS_RWBO	ACTVT	01 (Create or generate) 02 (Change) 03 (Display)
	DOMAIN	*
	DESTSYS	*
	TTYPE	TRAN
S_TRANSPRT	ACTVT	01 (Create or generate) 02 (Change) 03 (Display)
	TTYPE	TRAN
S_DATASET	ACTVT	34 (Write)
	FILENAME	*
	PROGRAM	SAPLSABC SAPLSTRF
S_ADMI_FCD	S_ADMI_FCD	SCMD (Write)

In the **central check system**, you need the following user roles to use transaction ATC to perform custom code checks:

User Role	Description
SAP_SATC_ADMIN	Authorization for setting up ABAP Test Cockpit (ATC) for central quality checking

In addition, you need the following authorization object for importing the Simplification Database into the central check system:

Name of Authorization Object	Name of the Authorization Field	Value of the Authorization Field	Description
S_YCM	SYCM_AREA	SDB	Authorization for importing the Simplification Database
	ACTVT	UL	

2.1.1.3 Applying SAP Notes

Apply the following SAP Notes in the **checked system**:

- 2485231 - Remote ATC Checks of Modifications and Enhancements
- 2270689 - RFC Extractor for performing static checks
- 2190065 - ATC/CI: Remote Code Analysis - Object Provider Stub

Apply all the relevant SAP Notes that are mentioned in the following SAP Notes in the **central check system**:

- 2436688 - Recommended SAP Notes for using SAP S/4HANA custom code checks in ATC
- 2364916 - Recommended SAP Notes for using ATC to perform remote analysis

2.1.1.4 Configuring the Connection Between Central Check System and Checked System

To perform the custom code analysis, you need to configure the connection between the Central Check System and the checked system.

2.1.1.4.1 Configuring RFC Connection Between the Central Check System and the Checked System

The `Custom Code Migration` app performs the SAP S/4HANA custom code checks in the checked system in which the custom code to be analyzed is stored.

Prerequisites

The RFC user in the checked system requires the authorizations that are specified in SAP Note 2672703.

The RFC connection must be usable without a logon dialog.

Context

You want to configure the required connection from the back-end system of this SAP Fiori app to the checked system.

For more information, see Establishing a Connection Using a Destination (SM59).

Procedure

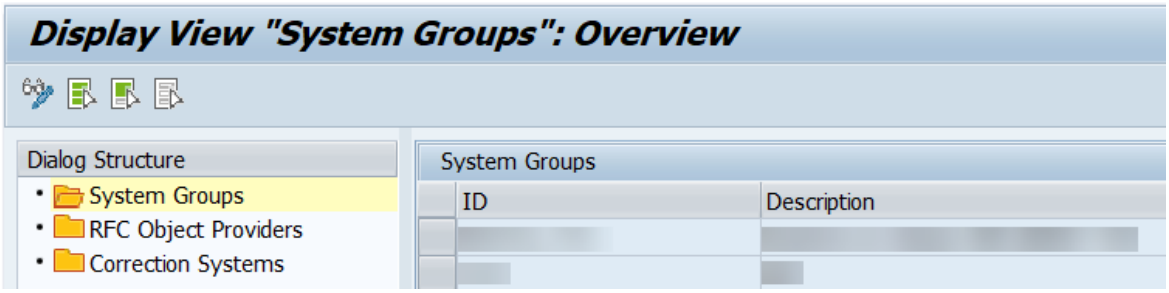
Use transaction `SM59` to create RFC connections for each checked system.

2.1.1.4.2 Creating System Groups

A system group is a logical unit that groups multiple SAP systems. Every object provider must be assigned to a system group and therefore you need to create a system group before you configure an object provider.

Procedure

1. Run transaction `ATC`.
2. In the navigation pane, expand the node ▶ *ATC Administration* ▶ *Setup* ▶ and double-click *Object Providers*.
3. Double-click the item *System Groups* for selection.



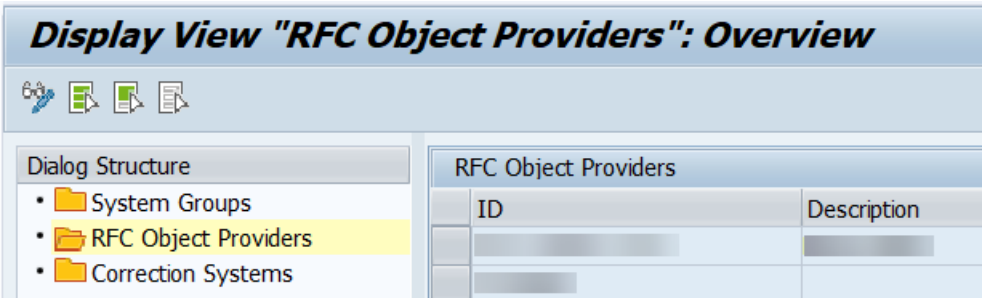
4. Switch to change mode (F6).
5. Choose the *New Entries* button in the toolbar.
6. Enter an *ID* and short *Description* for the new system group you want to add.
7. Save the new entry.

2.1.1.4.3 Configuring Object Providers

An object provider defines the RFC connection to be used for analysis in a remote SAP system. While a check run is being executed, the ATC framework uses this RFC connection to the checked system to extract a model from the custom code.

Procedure

1. In the *RFC Object Providers* view, double-click the item *RFC Object Providers* for selection.



2. Switch to change mode (F6).
3. Choose the *New Entries* button in the toolbar.
4. Specify the following entries for the object provider you want to create:

Fields	Description
`<ID>`	ID that specifies the object provider **Note** This ID is used when configuring an ATC run series.
`<Description>`	Short text that specifies the object provider
`<System Group>`	The system group to which the object provider belongs
`<SAP System>`	ID of the remote system, to which you want to connect using the object provider

Fields	Description
`<RFC Destination>`	Valid RFC destination for RFC connection to the checked system **Note** This RFC connection must be usable without a logon dialog. The RFC user in the checked system requires the authorizations that are specified in SAP Note 2672703.

5. Save the new entry.

2.1.1.5 Implementing the Custom Code Migration App

Perform the following tasks to implement the Custom Code Migration app.

Prerequisites

Before implementing the Custom Code Migration app, ensure that you have followed the steps for Implementing SAP Fiori Apps for Administrative, Configuration, and Extensibility Tasks.

Front-End Server: Enabling App for Access in SAP Fiori Launchpad

1. In transaction `PFCG`, add the business catalog `SAP_BASIS_BC_CCM` (Custom Code Migration) to the user roles that will use the `Custom Code Migration` app.

   > **Note**
   > This can be either an existing or a new user role.

2. Assign the user role to all users that will use the `Custom Code Migration` app.

The app is now available in SAP Fiori launchpad for all users that have assigned the changed user role.

> **→ Tip**
> Each user can add the app to its SAP Fiori launchpad by using the app finder in their user settings.

Front-End Server: Activating OData Services

1. In transaction `/IWFND/MAINT_SERVICE`, add the following technical service names as services:

Component	External Technical Name
Custom code migration projects	`SYCM_APS_C_PROJECT_CDS`
Analysis of SAP S/4HANA custom code check findings	`SYCM_APS_C_ATC_FIND_ALP_CDS`
Custom code scoping	`SYCM_APS_SCOPING`
Download of SAP S/4HANA custom code check findings	`SYCM_APS_FILE_SRV`
Access to the application log	`APL_LOG_MANAGEMENT_SRV`

For more information about activating OData services, see:

http://help.sap.com/s4hana `<Your on-premise edition>` ▶ *Discover* > *Product Assistance* > *Enterprise Technology* > *SAP Fiori* > *SAP Fiori Overview* > *Implement SAP Fiori Apps* > *UI-Specific Implementation Tasks* > *Activating OData Services* ◀

http://help.sap.com/fiori_implementation ▶ *Implementation* > *SAP Fiori: App Implementation* ◀.

Back-End Server: Assigning Authorizations

1. Assign the user role `SAP_BC_YCM_APS` to the users who will work with the `Custom Code Migration` app.

> **Note**
>
> For users who should only display the findings, you can add the authorization object `S_YCM_APS` to their roles and set the field `ACTVT` to the value `Display`.

2.1.2 Preparing the Custom Code Analysis

The following steps are relevant for the custom code analysis.

2.1.2.1 Importing the Simplification Database

To check your custom code against the SAP S/4HANA simplifications, you need to import the simplification database into the central check system.

2.1.2.1.1 Downloading the Simplification Database

SAP provides content for the simplification database of an SAP product in the SAP Support Portal. The content of the simplification database can be downloaded as a ZIP file. This ZIP file then has to be uploaded to the central check system.

Procedure

1. Open the SAP Software Download Center.
2. Choose *Types of Software* > *Installations & Upgrades* > *Access downloads* and search for **CCMSIDB**.
3. Choose the relevant ZIP file and choose *Download Basket*.

> **Note**
>
> To find the most recent content of the simplification database provided for SAP S/4HANA, see also SAP Note 2241080.

Results

The ZIP file is downloaded and saved on your selected drive. It contains the content of the simplification database.

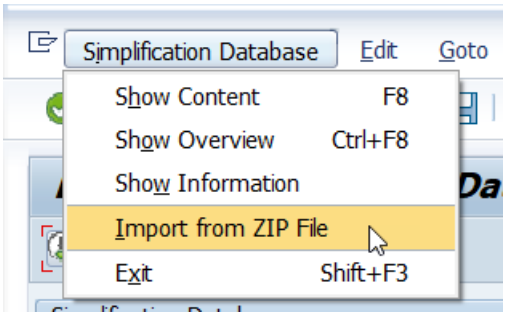
2.1.2.1.2 Importing the Simplification Database

To add the content of the simplification database to your system for further analysis, you need to upload the downloaded ZIP file to the central check system.

Procedure

1. Log on to the central check system.
2. Run transaction `SYCM`.

   The *Display simplification database Content* view is opened.
3. Choose *simplification database* > *Import from ZIP File* from the menu bar.



4. Choose the ZIP file with the downloaded simplification database from your drive and confirm with *Open*.

Results

The simplification database is uploaded to the central check system and is now available for analysis in transaction `SYCM`.

2.1.2.1.3 Displaying the Content of the Simplification Database

You can display the content of the simplification database to get an overview of all changes or specific changes and the corresponding SAP objects that are simplified in SAP S/4HANA.

Procedure

To get a list of all simplification items, choose `Ctrl` + `F8` or the *Overview* button.

> **Note**
>
> To limit the number of simplification items displayed, enter the relevant filter criteria in transaction *SYCM* and choose Execute.

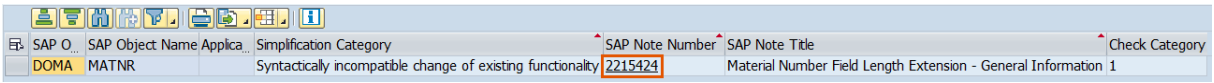
Results

The *simplification database Content* view is opened. From here you can order the displayed list, for example by their *SAP Object Type*, the assigned *Simplification Category*, or the relevant *SAP Note Number*.

From the overview, you can display the list of objects contained in a simplification item.

Example

To find out whether a certain object is related to a simplification item, enter the name of the object (for example **MATNR**) in the *Object Name* field and execute (`F8`). Then you can navigate to the corresponding SAP Note to get more information.



SAP O...	SAP Object Name	Applica...	Simplification Category	SAP Note Number	SAP Note Title	Check Category
DOMA	MATNR		Syntactically incompatible change of existing functionality	2215424	Material Number Field Length Extension - General Information	1

2.1.3 Custom Code Analysis with SAP Fiori App (Recommended)

The `Custom Code Migration` app enables you to scope and analyze custom code that needs to be migrated from an SAP Business Suite system to SAP S/4HANA. In addition, this app supports you with identifying unused custom code based on your collected usage data. This enables you to remove unused custom code during the system conversion.

Key Features:

Analysis of custom code:

- Graphical representation of custom code analysis results
- Results can be filtered by various categories, such as quick fix availability, scope and usage data.

Scoping:

- Based on usage data, you can define which ABAP custom code needs to be migrated to SAP S/4HANA.
- This app creates a deletion transport in order to delete unused ABAP source code during the system conversion to SAP S/4HANA.

The `Custom Code Migration` app enables you to create an SAP S/4HANA Migration Project.

With the help of an SAP S/4HANA Migration Project, you can analyze custom code that shall be migrated from an existing product like SAP Business Suite to SAP S/4HANA.

> **Note**
>
> In addition to the SAP S/4HANA Migration Project, the `Custom Code Migration` app offers two additional project types.
>
> - The SAP BTP Analysis Project enables you to analyze custom code for readiness to run in SAP BTP ABAP environment.
> - The Custom Code Analysis Project enables you to analyze custom code with arbitrary ATC check variants.
>
> These project types are not relevant for the custom code migration process and will not be described in this guide. To find out more about the SAP BTP Analysis and Custom Code Analysis project, you can use the in-app help and the guided tours in the `Custom Code Migration` app and/or consult the Custom Code Analysis SAP Help Portal documentation.

2.1.3.1 Creating a Custom Code Migration Project

A custom code migration project defines the scope of custom code you want to analyze and migrate to SAP S/4HANA.

Procedure

1. Select the tile *Custom Code Migration* from the SAP Fiori launchpad.
2. Choose *Create* to create a new custom code migration project.
3. Enter a *Project Description* for your new project.
4. Define the *Transition Scenario*, the *Target Release* of the back-end and the *Material Number Length*.

   > ⓘ Note
   >
   > Make sure that the checkboxes for *Extended Amount*, *Extended Stock/Requirement Segment* and *Extended Season/Theme/Collection* are enabled, if you intend to use them.

5. In the *System to Be Analyzed* section, define the *Location of Custom Code* and the *Object Provider to Remote System*. You define the object provider in the communication scenario.
6. In the *Advanced Configuration* tab, under *Analysis Configuration*, you can either leave the check box *Start Analysis at Project Creation* checked and start the custom code analysis automatically at project creation or uncheck it and start the analysis manually at a later point.
7. To confirm your input, choose *Create*.

2.1.3.2 Defining the Scope of Your Custom Code Migration Project

The Custom Code Migration app supports you with identifying your unused custom code based on your collected usage data. With the scope of a custom code migration project, you can specify which of your custom code needs to be converted to SAP S/4HANA.

To define the scope of your custom code migration project, you can either use your usage data collected with transaction SUSG or your usage data collected in your Solution Manager (this requires Solution Manager 7.2 support package 8 or higher).

> ⓘ Note
>
> If you have not aggregated your usage data so far, you should activate the aggregation of your usage data in transaction **SUSG** in your production system now. To do so, start transaction **SUSG** in your production system and choose *Activate* to activate the aggregation of your usage data.
>
> If you have applications that you only use in your development or test system, we recommend that you also collect usage data from these systems.
>
> Ideally, the usage data you add to your custom code migration project should cover at least one year of usage information, so that also usage data of quarter and year ending functionality is considered.

2.1.3.2.1 Adding Usage Data Stored in SUSG

To define the scope of your custom code migration project, you can use your usage data collected with transaction SUSG.

Prerequisites

> **Note**
>
> If you have applications that you only use in your development or test system, we recommend that you also add usage data from these systems.

Procedure

1. Start transaction SUSG in your production system.

   The *Usage Data: Aggregation State* view is opened.
2. Choose *Create Snapshot* to create a snapshot of your usage data.
3. Recommended: Transfer your usage data from your production system to your checked system.
   a. Choose *Manage Snapshots*.
   b. Choose *Download to File* to export your snapshot as a file.
   c. Choose *Manage Snapshots* in transaction SUSG in your checked system.
   d. Choose *Upload from File* to import your usage data from a file.
4. Add the snapshot of your usage data to your custom code migration project in the `Custom Code Migration` app.
   a. Choose *Edit*.
   b. In the *Usage Data* frame, choose *Add*.
   c. Choose beneath the *Usage Description* field.
   d. Search for your usage data you created as a snapshot.

      > **Note**
      >
      > If you transferred your usage data to your checked system, your usage data is displayed automatically in the *Items* overview.

   e. Select your usage data to add it to your custom code migration project.

   > **Note**
   >
   > When you add usage data to your custom code migration project, a default scope is calculated. The default scope includes all used objects, all objects which are statically referenced by the used objects, and objects for which no usage data is available, such as database tables or data elements.
5. Choose *Scope* to display the scope information.

Here you can see how many objects are in and out of scope.

> → Tip
>
> Choose *View By* to display detailed scope information or your scope sorted by object name or object type.

2.1.3.2.2 Adding Usage Data Stored in SAP Solution Manager

You can also upload your usage data collected in the SAP Solution Manager (using UPL or SCMON).

Prerequisites

Your SAP Solution Manager system must be on the release 7.20 (SP8) or higher.

The following SAP Notes must be implemented:

- 2599695
- 2190065

On the central check system, the RFC destination to the SAP Solution Manager system must be created in the transaction SM59. See Configuring the Connection Between Central Check System and Checked System [page 14].

> ⓘ Note
>
> If you have applications that you only use in your development or test system, we recommend that you also add usage data from these systems.

Procedure

1. Add the usage data to your custom code migration project in the `Custom Code Migration` app.
   a. Choose *Edit*.
   b. In the *Usage Data* frame, choose *Add*.
2. Use the value help in the *Destination for Usage* field.
3. Enter the destination to your SAP Solution Manager system in the *Connection* field and choose your destination in the corresponding table row.
4. In the *Usage Description* field, use the value help to select the right SAP Solution Manager usage data snapshot from the list.

Results

The Solution Manager usage data will be added to your custom code migration project.

> **Note**
>
> When loading usage data collected in SAP Solution Manager, the usage counter will not display usage numbers (0 usages). Usage counter data is not supported by the APIs of SAP Solution Manager.
>
> You can find the usage counter here: *Scope* > *Change Scope* > *Entry Points (List Page)* > *Choose a Request Entry Point* > *Procedures within the Entry Point* > *Procedures* > *Number Usages*.

2.1.3.2.3 Changing the Scope of Your Custom Code Migration Project

In the Custom Code Migration app, you can change the scope on the following levels:

- **Request Entry Points**
  Request entry points are all applications or services which are used in your system, such as programs, transactions, URLs, etc. For each request entry point, the used objects are recorded.
  In the initial scope, all used request entry points are added to the scope. You can remove request entry points from the scope by executing the following steps:
  1. Choose *Scope*.
  2. Choose *Change Scope*.
  3. Choose *Entry Points* (list page).
  4. Select the request entry point you want to remove from the scope.
  5. Choose *Remove Entry Point from Scope*.
- **Packages**
  You can add or remove whole packages to/from the scope by executing the following steps:
  1. Choose *Scope*.
  2. Choose *Change Scope*.
  3. Choose *Packages* (list page).
  4. Select the package you want to add to or remove from the scope.
  5. Choose *Add Package to Scope* or *Remove Package from Scope*.
- **Objects**
  You can add or remove single objects to/from the scope by executing the following steps:
  1. Choose *Scope*.
  2. Choose *Change Scope*.
  3. Choose *Packages* (list page).
  4. Navigate to the package of the objects.
  5. Select the object you want to add to or remove from the scope either in the list “Scoped Objects” or in the list “Not Scoped Objects”.
  6. Choose *Add Object to Scope* or *Remove Object from Scope*.

When is an object in scope or out of scope?

Objects are in scope, if ...

- ... they are added to the scope explicitly (by object),
- ... their package is added to scope explicitly,
- ... they are used by at least one request entry point which is in scope,
- ... they are referenced by any other object which is in scope,
- or their object type has no usage data.

Objects are out of scope, if ...

- ... they are removed from scope explicitly (by object),
- or they are not used by any request entry point.

Example

When do you need to change the scope of your custom code migration project?

- **Scenario 1 – Remove transaction from scope**
  A transaction that is used in your Business Suite system shall not be used any longer, because you want to change your business processes in SAP S/4HANA.
  Choose the filter *Scope by Request Entry Points* for your transaction and remove the request entry point from your scope.
- **Scenario 2 – Add new development**
  A new functionality is in development but has not been used in your production system yet. Therefore, no usage data has been collected for this application and it has not been added to your scope automatically.
  Choose the filter *Scope by Package* for all packages containing your developments for the new functionality and add them to your scope.
- **Scenario 3 – Remove complexity**
  Sometimes, your ABAP code can contain complex objects that very often also experience the most frequent changes. With the `Custom Code Migration` app it is possible to identify the most complex objects via complexity analysis and help you decide whether you want to redesign and therefore simplify them.
  In your custom code migration project, choose the bubble chart to display all packages that contain custom code. With this chart, you can also display the Halstead Difficulty of a package. The Halstead complexity measures are software metrics that describe the effort to develop and understand the complexity of a program.

2.1.3.2.4 Removing Unused Custom Code During System Conversion

When you have specified your scope in your custom code migration project, you can create a deletion transport in the checked system. This deletion transport contains all objects which will not be migrated during the system conversion.

Procedure

Choose *Create Deletion Transport* to create the deletion transport.

The deletion transport is also shown in the field *Request/Task* in your custom code migration project.

Next Steps

For more information, see chapter "Integration of Customer Transports" in the corresponding upgrade guide for SAP HANA DB Software Update Manager 2.0.

2.1.3.3 Analyzing the Findings

When you create a custom code migration project, the SAP S/4HANA custom code checks are performed automatically. The Custom Code Migration app then gives you an analytical representation of SAP S/4HANA custom code check findings.

Procedure

1. Choose *Analysis* to get an overview of the results, as soon as the first analysis has been finished.
2. Choose *Analyze Findings* to get a detailed analysis.
3. In the *Custom Code Migration - Analysis - Findings* view, you can specify various filters, for example:
   - *Simplification Item Category*: The simplification item category specifies whether findings are related to functionality which is not available any more or has been changed in an incompatible way.
   - *SAP Note Number*: Lets you analyze findings of a specific Simplification Item (for example field length extension of material number)
   - *Scope Information*: Lets you filter the findings to show only findings for objects that are in scope.
   - *Quick Fix Availability*: Lets you analyze which findings can be solved by a Quick Fix and which findings have to be solved manually.
4. Choose ⚙ to change the settings for the chart.

   To show the Quick Fix availability per SAP Note title, you must specify the following settings for the chart:

- SAP Note title (as Category)
- Findings (as Axis 1)
- Quick Fix Availability (as Series)

5. [Optional:] If you're leveraging AI capabilities in this workflow, please see Working with the Explain Capability In-App.

Results

You can now continue with chapter Custom Code Adaption Before System Conversion [page 62].

> **Note**
>
> It is recommended to focus on Priority 1 and 2 findings first. See also SAP Note 3735015 for adaptions to priorities for SAP S/4HANA custom code checks.

2.2 Using SAP NetWeaver Application Server for ABAP 7.52 as Central Check System

Performing the Custom Code Analysis

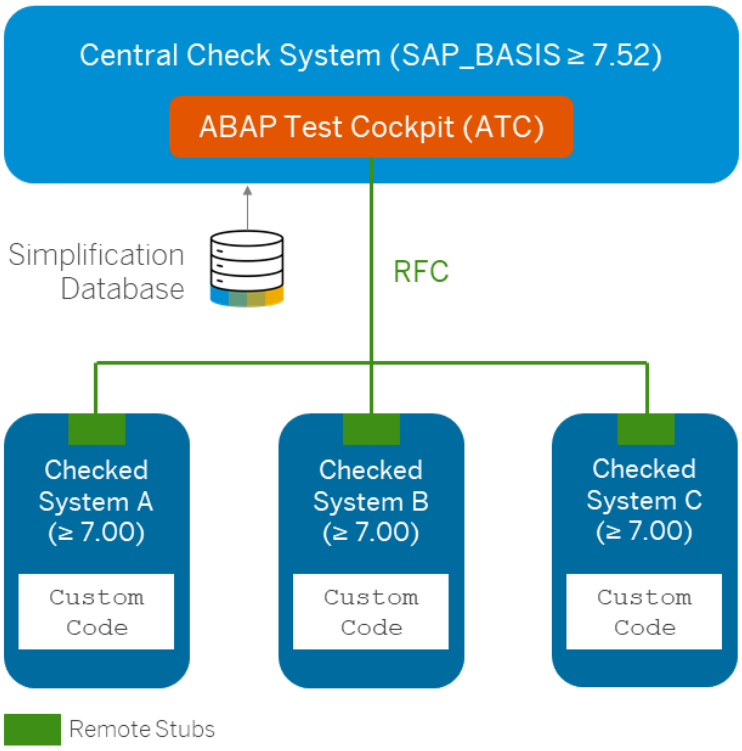
If the central check system is based on SAP NetWeaver Application Server for ABAP 7.52, your custom code needs to be checked remotely against the SAP S/4HANA simplification items in the simplification database using ABAP Test Cockpit (ATC) in the classic SAP GUI. The result is a list of findings where your custom code does **not** comply with the scope and data structure of SAP S/4HANA. At this point, you can estimate the effort required to adapt custom code to be migrated to SAP S/4HANA.

> **Note**
>
> The central check system can be used to check one or more SAP Business Suite systems.

Enabling the Custom Code Analysis

To perform the custom code analysis, you need to set up a centralized SAP NetWeaver AS for ABAP 7.52 system as the central check system within your SAP system landscape.

For remote access, the central check system needs RFC destinations for each relevant system that you want to check using ATC. When executed, the central check system accesses the checked system using remote stubs and the RFC connection. These remote stubs are an interface between the central check system and the checked system and return a model from custom code that needs to be checked.



2.2.1 Getting Started

2.2.1.1 System Requirements

To analyze your custom code in SAP GUI, you need to meet the following system requirements:

SAP System	Software Component
Central Check System	`SAP_BASIS 7.52` or higher
Checked System	SAP Business Suite system based on `SAP_BASIS 7.00` or higher

2.2.1.2 Configuring the User

You can either apply SAP Note 3548772 in the **checked system** to get all relevant authorizations for your RFC user or, alternatively, check the table below.

Authorization Object Value	Parameter	Value
S_RFC	ACTVT	16 (Execute)
	RFC_TYPE	FUGR
	RFC_NAME	SCA_REMOTE_DATA_ACCESS SABP_COMP_PROCS_E SYCM_APS_REMOTE SYST S_CODE_INSPECTOR_TESTS
S_RFC	ACTVT	16 (Execute)
	RFC_TYPE	FUNC
	RFC_NAME	FUNCTION_EXISTS REPOSITORY_ENVIRONMENT_ALL RFC_GET_NAMETAB SVRS_GET_VERSION_DIRECTORY_46 RFCPING SLINRMT_RUN TRINT_PROGRESS_INDICATOR TRINT_TP_UPDATE_TPSTAT SLDAG_GET_SYSTEM_NAME CVA_BSP_RUN IDOCTYPE_COLLECT_ATC_INFOS IQAPI_QUERY_CONTENT IQAPI_INFOSET_CONTENT IQAPI_CHECK_INFOSET FM_CCM_DYNPRO_READ SCI_GET_DDLS
S_DEVELOP	ACTVT	03 (Display)
	DEVCLASS	*
	OBJNAME	*
	OBJTYPE	*
	P_GROUP	*

Authorization Object Value	Parameter	Value
S_SYS_RWBO	ACTVT	01 (Create or generate) 02 (Change) 03 (Display)
	DOMAIN	*
	DESTSYS	*
	TTYPE	TRAN
S_TRANSPRT	ACTVT	01 (Create or generate) 02 (Change) 03 (Display)
	TTYPE	TRAN
S_DATASET	ACTVT	34 (Write)
	FILENAME	*
	PROGRAM	SAPLSABC SAPLSTRF
S_ADMI_FCD	S_ADMI_FCD	SCMD (Write)

2.2.1.3 Applying SAP Notes

Apply the following SAP Notes in the **checked system**:

- 2485231 - Remote ATC Checks of Modifications and Enhancements
- 2270689 - RFC Extractor for performing static checks
- 2190065 - ATC/CI: Remote Code Analysis - Object Provider Stub

Apply all the relevant SAP Notes that are mentioned in the following SAP Notes in the **central check system**:

- 2436688 - Recommended SAP Notes for using SAP S/4HANA custom code checks in ATC
- 2364916 - Recommended SAP Notes for using ATC to perform remote analysis

2.2.1.4 Configuring the Connection Between Central Check System and Checked System

To perform the custom code analysis, you need to configure the connection between the Central Check System and the checked system.

2.2.1.4.1 Configuring RFC Connection Between the Central Check System and the Checked System

The `Custom Code Migration` app performs the SAP S/4HANA custom code checks in the checked system in which the custom code to be analyzed is stored.

Prerequisites

The RFC user in the checked system requires the authorizations that are specified in SAP Note 2672703.

The RFC connection must be usable without a logon dialog.

Context

You want to configure the required connection from the back-end system of this SAP Fiori app to the checked system.

For more information, see Establishing a Connection Using a Destination (SM59).

Procedure

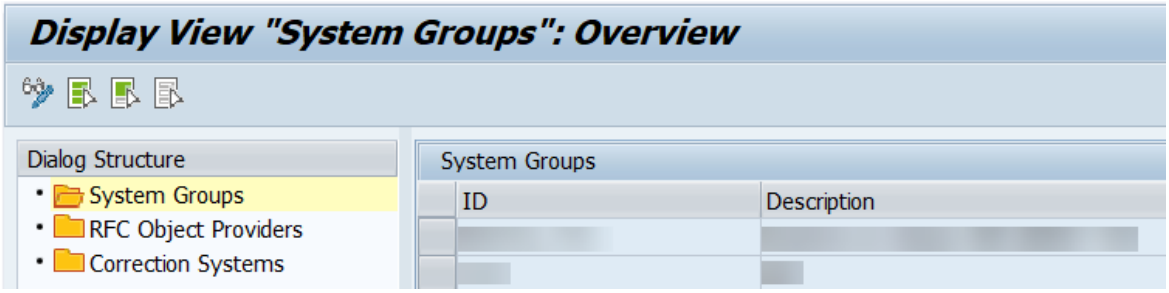
Use transaction `SM59` to create RFC connections for each checked system.

2.2.1.4.2 Creating System Groups

A system group is a logical unit that groups multiple SAP systems. Every object provider must be assigned to a system group and therefore you need to create a system group before you configure an object provider.

Procedure

1. Run transaction `ATC`.
2. In the navigation pane, expand the node *ATC Administration* > *Setup* and double-click *Object Providers*.
3. Double-click the item *System Groups* for selection.



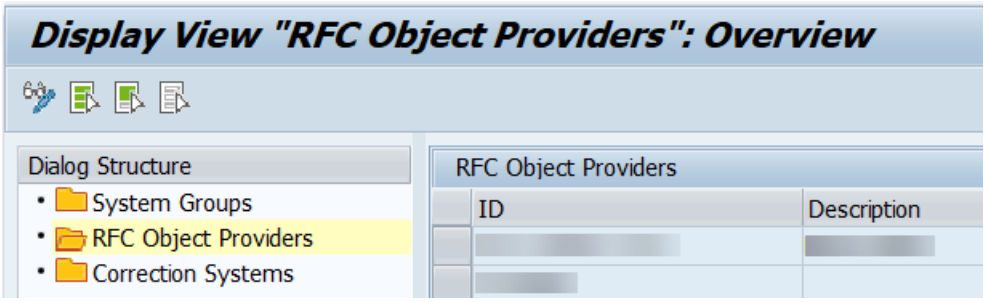
4. Switch to change mode (F6).
5. Choose the *New Entries* button in the toolbar.
6. Enter an *ID* and short *Description* for the new system group you want to add.
7. Save the new entry.

2.2.1.4.3 Configuring Object Providers

An object provider defines the RFC connection to be used for analysis in a remote SAP system. While a check run is being executed, the ATC framework uses this RFC connection to the checked system to extract a model from the custom code.

Procedure

1. In the *RFC Object Providers* view, double-click the item *RFC Object Providers* for selection.



2. Switch to change mode (F6).
3. Choose the *New Entries* button in the toolbar.
4. Specify the following entries for the object provider you want to create:

Fields	Description
`<ID>`	ID that specifies the object provider **Note** This ID is used when configuring an ATC run series.
`<Description>`	Short text that specifies the object provider
`<System Group>`	The system group to which the object provider belongs
`<SAP System>`	ID of the remote system, to which you want to connect using the object provider
`<RFC Destination>`	Valid RFC destination for RFC connection to the checked system **Note** This RFC connection must be usable without a logon dialog. The RFC user in the checked system requires the authorizations that are specified in SAP Note 2672703.

5. Save the new entry.

2.2.2 Preparing the Custom Code Analysis

The following steps are relevant for the custom code analysis.

2.2.2.1 Importing the Simplification Database

To check your custom code against the SAP S/4HANA simplifications, you need to import the simplification database into the central check system.

2.2.2.1.1 Downloading the Simplification Database

SAP provides content for the simplification database of an SAP product in the SAP Support Portal. The content of the simplification database can be downloaded as a ZIP file. This ZIP file then has to be uploaded to the central check system.

Procedure

1. Open the SAP Software Download Center.
2. Choose *Types of Software* > *Installations & Upgrades* > *Access downloads* and search for **CCMSIDB**.
3. Choose the relevant ZIP file and choose *Download Basket*.

> **Note**
>
> To find the most recent content of the simplification database provided for SAP S/4HANA, see also SAP Note 2241080.

Results

The ZIP file is downloaded and saved on your selected drive. It contains the content of the simplification database.

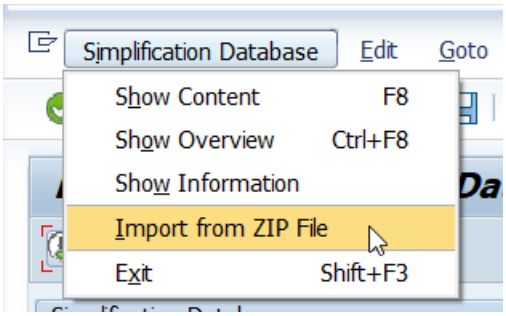
2.2.2.1.2 Importing the Simplification Database

To add the content of the simplification database to your system for further analysis, you need to upload the downloaded ZIP file to the central check system.

Procedure

1. Log on to the central check system.
2. Run transaction `SYCM`.

   The *Display simplification database Content* view is opened.
3. Choose *simplification database* > *Import from ZIP File* from the menu bar.



4. Choose the ZIP file with the downloaded simplification database from your drive and confirm with *Open*.

Results

The simplification database is uploaded to the central check system and is now available for analysis in transaction `SYCM`.

2.2.2.1.3 Displaying the Content of the Simplification Database

You can display the content of the simplification database to get an overview of all changes or specific changes and the corresponding SAP objects that are simplified in SAP S/4HANA.

Procedure

To get a list of all simplification items, choose `Ctrl` + `F8` or the *Overview* button.

> ⓘ Note
>
> To limit the number of simplification items displayed, enter the relevant filter criteria in transaction *SYCM* and choose Execute.

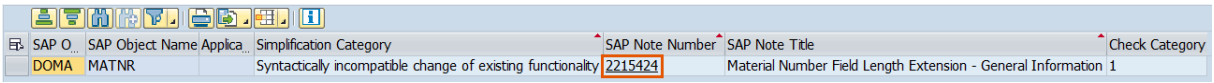
Results

The *simplification database Content* view is opened. From here you can order the displayed list, for example by their *SAP Object Type*, the assigned *Simplification Category*, or the relevant *SAP Note Number*.

From the overview, you can display the list of objects contained in a simplification item.

Example

To find out whether a certain object is related to a simplification item, enter the name of the object (for example **MATNR**) in the *Object Name* field and execute (`F8`). Then you can navigate to the corresponding SAP Note to get more information.



SAP O...	SAP Object Name	Applica...	Simplification Category	SAP Note Number	SAP Note Title	Check Category
DOMA	MATNR		Syntactically incompatible change of existing functionality	2215424	Material Number Field Length Extension - General Information	1

2.2.3 Custom Code Analysis in SAP GUI

If the central check system is based on SAP NetWeaver Application Server for ABAP 7.52, your custom code needs to be checked remotely against the SAP S/4HANA simplification items in the simplification database using the ABAP Test Cockpit (ATC) in the classic SAP GUI.

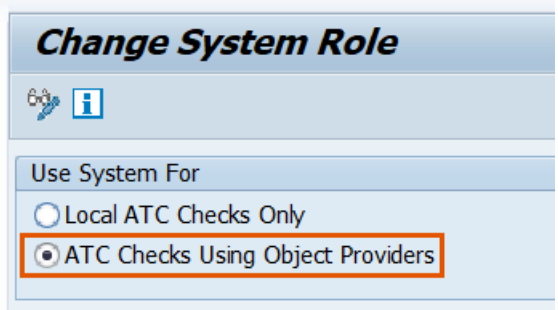
2.2.3.1 Setting the Role as Central Check System

The system role needs to be specified as the central check system for remote SAP S/4HANA checks.

Procedure

1. Run transaction ATC.

   The *ABAP Test Cockpit Overview* screen appears.
2. In the navigation pane, expand the node *ATC Administration* > *Setup* and double-click *System Role*.
3. Switch to change mode (F6).
4. In the *Change System Role* view, choose *ATC Checks Using Object Providers*.



5. Save the new settings and return to the *ABAP Test Cockpit Overview* screen.

2.2.3.2 Running Remote SAP S/4HANA Checks

In the Central Check System, you can perform SAP S/4HANA checks to analyze development objects in a checked system remotely. In this step, you can estimate the effort required to adapt your ABAP source code to SAP S/4HANA-related changes.

2.2.3.2.1 Configuring a Remote ATC Run Series

Procedure

1. Run transaction `ATC`.
2. In the navigation pane, expand the nodes *ATC Administration* > *Runs* and double-click *Schedule Runs*. The screen that appears displays a list of existing run series.
3. Choose the *Create* button in the toolbar.
4. Enter the name for the series you want to create and confirm.
5. Specify the following entries for the run series you want to create:

Field	Description
`<Description>`	Enter a short description for the ATC run series you want to configure. → Tip You can use the template provided by the *Description* field. When you execute the run series, the built-in variables are filled with data for the system, the day of the week, the calendar week, and for the year. However, you can also add further texts to these variables, rearrange them, or replace the built-in variables with your own text.

Field	Description

`<Check Variant>`	Enter one of the following check variants:
	`S4HANA_READINESS_2025_NO_FLE`
	Predefined settings:
	• Material number length = **18**
	• Extended amount field length is **disabled**
	• Extended stock requirement field length is **disabled**
	• Extended season/theme/collection field length is **disabled**
	`S4HANA_READINESS_2025`
	Predefined settings:
	• Material number length = **40**
	• Extended amount field length is **enabled**
	• Extended stock requirement field length is **enabled**
	• Extended season/theme/collection field length is **enabled**
	ⓘ Note
	These product-specific global check variants check all the simplifications relevant for SAP S/4HANA 2025, including those for previous releases.

`<Object Provider>`	Enter the ID of the relevant object provider that represents the remote system you want to check.
`<Objects to Check>`	Specify the set of development objects you want to check in the checked system. On the *Checkable Namespaces* tab, the option *By Query* lets you specify objects by name (for example, by package name), by transport layer, or by component. The option *By Object Set* lets you specify an object set you defined in Code Inspector in the checked system. → Tip You can use the value help to choose the packages or the object set in the remote system. On the *Modified Objects* tab, you can specify the modified source code objects you want to check.

6. *Save* the configuration.

2.2.3.2.2 Scheduling a Remote ATC Run Series

Procedure

1. In the list of run series, select the run series in question and click the *Schedule* button in the toolbar.
2. In the dialog that appears, adapt the settings for the *Life Span* of the series and for *Execution* to your needs. Otherwise, leave these settings unchanged.

   > **Note**
   >
   > The *Life Span* defines how long (in days) the ATC result is kept in the system. After this, the ATC result is automatically deleted from the system.

3. Choose *Execute* (F8).

   > **Note**
   >
   > You can use the option *Execute in Background* (F9) to schedule a run series at regular intervals.

2.2.3.2.3 Monitoring an ATC Run Series

In this step, you can monitor the status of the remote ATC run series.

Procedure

1. Run transaction ATC.
2. In the navigation pane, expand the nodes *ATC Administration* > *Runs* and double-click *Monitor and Control Runs*.
3. Leave the run series field empty and execute (F8).

   The *ATC Run Monitor* view is displayed. The view shows the status of check runs as running, finished, or failed.

2.2.3.2.4 Analyzing ATC Run Results

Procedure

1. Run transaction ATC.
2. In the navigation pane, expand the nodes *ATC Administration* > *Runs* and double-click *Manage Results*.

3. Leave the run series field empty and choose *Execute* (F8).

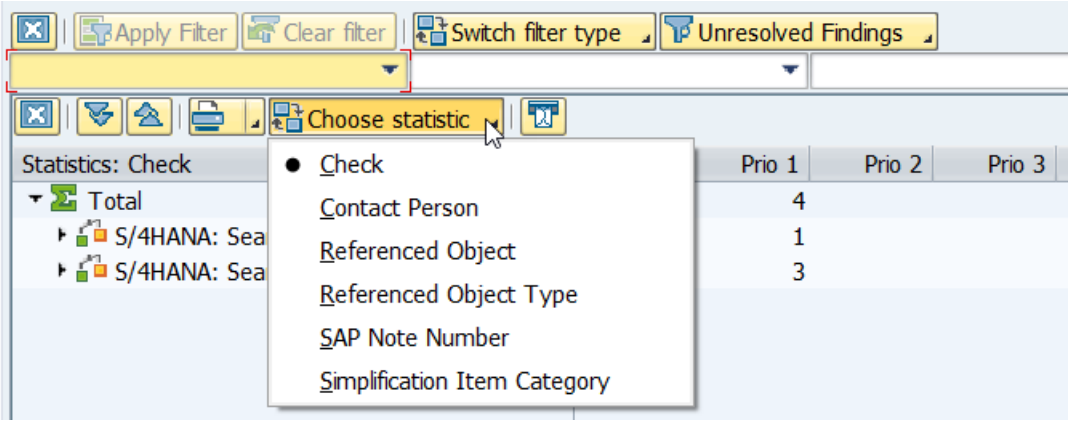
> **Note**
>
> By default, the ATC run results of the last 10 days are displayed. If you want to see the ATC run results from a different period of time, change the settings in *Schedule Data*.

The *ATC Manage Results* view is displayed. The view shows a list of ATC run results from your requested period of time.

4. In the *ATC Manage Results* view, select a run series and choose *Display* to inspect the results.

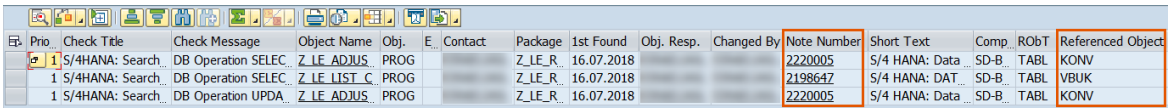
   A list of all ATC findings is displayed in transaction `SE80`.

5. Choose *Statistics View*.
6. Choose *Choose Statistic* to group the ATC findings (for example, by SAP Note Number).



   The ATC findings are grouped by the selected statistics.

7. Double-click a statistics group to display the list of ATC findings assigned to the selected statistics.
8. Double-click an ATC finding to open detailed information. Here you can find the SAP Note number referring to a dedicated SAP Note and the referenced object.



Prio	Check Title	Check Message	Object Name	Obj.	E	Contact	Package	1st Found	Obj. Resp.	Changed By	Note Number	Short Text	Comp	RObT	Referenced Object
1	S/4HANA: Search	DB Operation SELEC	Z_LE_ADJUS	PROG			Z_LE_R	16.07.2018			2220005	S/4 HANA: Data	SD-B	TABL	KONV
1	S/4HANA: Search	DB Operation SELEC	Z_LE_LIST_C	PROG			Z_LE_R	16.07.2018			2198647	S/4 HANA: DAT	SD-B	TABL	VBUK
1	S/4HANA: Search	DB Operation UPDA	Z_LE_ADJUS	PROG			Z_LE_R	16.07.2018			2220005	S/4 HANA: Data	SD-B	TABL	KONV

9. Click the object name to navigate to the source code to see where the incompatible code occurs.

Results

You can now continue with chapter Custom Code Adaptation After System Conversion [page 64].

2.3 Using SAP BTP ABAP environment as Central Check System (Recommended Solution by SAP)

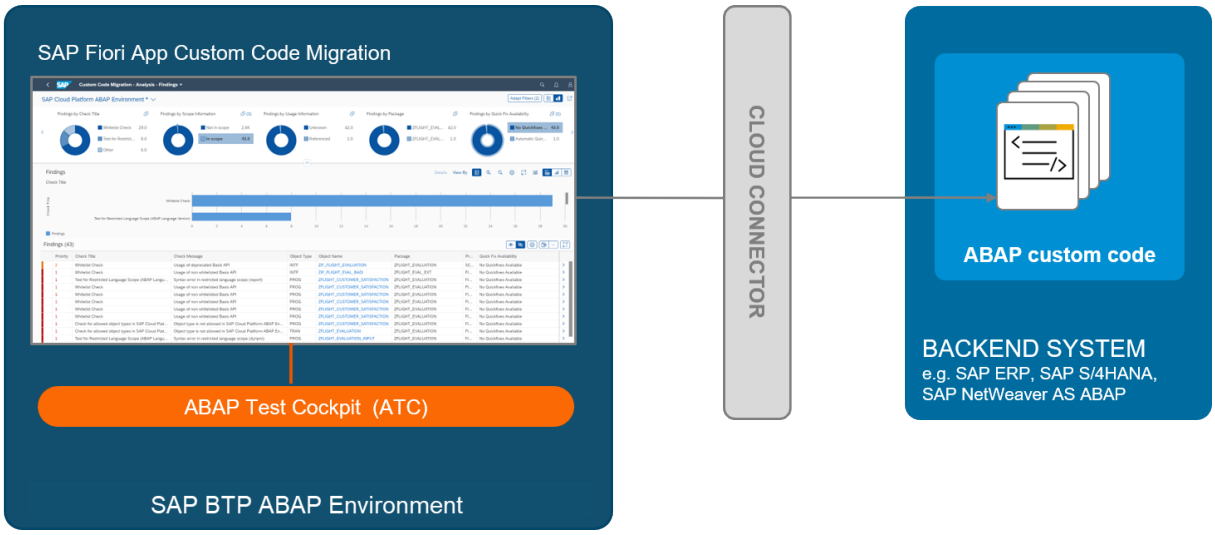
If you use **SAP BTP ABAP environment** as central check system, you will use the Custom Code Migration app to scope and analyze custom code in an on-premise SAP system for SAP S/4HANA.

You want to bring your custom code from an SAP Business Suite system to SAP S/4HANA. The central check system, where the analysis for possible adaptations takes place, is based on SAP BTP ABAP environment. Note that in this case always the latest ATC checks are provided.

To bring your custom code from an SAP Business Suite system to SAP S/4HANA, the `Custom Code Migration` app enables you to find the relevant parts in your source code to estimate the effort for converting your customer code to SAP S/4HANA, on-premise edition.

To use this SAP Fiori app, your SAP administrator must enable the app to provide access for the relevant users. In addition, you also need to establish an RFC connection for the Cloud Connector. This enables you to access the on-premise system from the ABAP environment remotely.

The connection between the ABAP environment as central check system to the on-premise system as Checked System is established through an RFC connection that is configured as destination and a communication arrangement in the Cloud Connector.



Remote infrastructure using an RFC connection between an on-premise SAP system and SAP BTP ABAP environment across the Cloud Connector

2.3.1 Automated Setup (Recommended)

We recommend using the Cloud Integration Automation Service (CIAS) to set up the Custom Code Migration app in the SAP BTP ABAP environment. This service provides a guided workflow for simpler and faster setup and configuration. You'll find more information about the CIAS in the Cloud Integration Automation Service User Guide.

> **Note**
>
> You can also do the setup manually as described in chapter Manual Setup [page 43].

To start the CIAS workflow, proceed as follows:

1. Verify that the prerequisites for the CIAS are fulfilled, as described in the CIAS user guide in the topic Pre-Requisites.
2. Subscribe to the Cloud Integration Automation Service, as described in Subscription to Standard Plan.
3. Assign the roles `CIASIntegrationExpert` and `CIASIntegrationAdministrator` to your user, as described in Assigning Roles to the Users.
4. Go to *Services* > *Instances and Subscriptions* > *Subscriptions* to check the status of the CIAS application. Once the status has changed to *Subscribed*, choose *(Go to Application)*.
5. The CIAS launchpad opens. Choose the tile *Plan for Integration* to display all available Solutions and Scenarios.
6. Use the *Search* field to find the *Custom Code Migration* scenario.
7. Open the scenario and choose *Select Systems*.
8. Select your on-premise SAP system and your SAP Cloud Connector.
9. Choose *Generate Workflow*, then follow the steps to set up the Custom Code Migration app.

2.3.2 Manual Setup

2.3.2.1 Prerequisites Check

Before you start enabling the Custom Code Migration app for SAP BTP ABAP environment, check whether the following prerequisites are fulfilled:

Accounts and Subaccounts

- You have a customer account for SAP Business Technology Platform
- You have an entitlement for the ABAP environment.
- You have received your logon data by email.
- You have administration authorizations to perform the steps to set up the ABAP environment.
- In the Cloud Foundry organization, you are a member with the organization manager role.

For more information about getting started with the ABAP environment, see the SAP BTP documentation on SAP Help Portal: Getting Started with a Customer Account: Workflow in the ABAP environment.

Setting up an ABAP environment

Before you can use the `Custom Code Migration` app in the context of SAP BTP ABAP environment, set up the ABAP environment for your company.

> → Recommendation
>
> SAP recommends you using a booster. See Using a Booster to Automate the Setup of the ABAP Environment (Optional). Using this booster, you have to perform the step Creating a Service Key (Optional) for the Custom Code Migration app in addition.

For more information about the general ABAP environment setup, see Getting Started in the ABAP environment

Cloud Connector

You have installed the Cloud Connector. For more information see Cloud Connector.

2.3.2.2 Configuring the User

You can either apply SAP Note 3548772 in the **checked system** to get all relevant authorizations for your RFC user or, alternatively, check the table below.

Authorization Object Value	Parameter	Value
S_RFC	ACTVT	16 (Execute)
	RFC_TYPE	FUGR
	RFC_NAME	SCA_REMOTE_DATA_ACCESS SABP_COMP_PROCS_E SYCM_APS_REMOTE SYST S_CODE_INSPECTOR_TESTS
S_RFC	ACTVT	16 (Execute)
	RFC_TYPE	FUNC

Authorization Object Value	Parameter	Value
	RFC_NAME	FUNCTION_EXISTS REPOSITORY_ENVIRONMENT_ALL RFC_GET_NAMETAB SVRS_GET_VERSION_DIRECTORY_46 RFCPING SLINRMT_RUN TRINT_PROGRESS_INDICATOR TRINT_TP_UPDATE_TPSTAT SLDAG_GET_SYSTEM_NAME CVA_BSP_RUN IDOCTYPE_COLLECT_ATC_INFOS IQAPI_QUERY_CONTENT IQAPI_INFOSET_CONTENT IQAPI_CHECK_INFOSET FM_CCM_DYNPRO_READ SCI_GET_DDLS
S_DEVELOP	ACTVT	03 (Display)
	DEVCLASS	*
	OBJNAME	*
	OBJTYPE	*
	P_GROUP	*
S_SYS_RWBO	ACTVT	01 (Create or generate) 02 (Change) 03 (Display)
	DOMAIN	*
	DESTSYS	*
	TTYPE	TRAN
S_TRANSPRT	ACTVT	01 (Create or generate) 02 (Change) 03 (Display)

Authorization Object Value	Parameter	Value
	TTYPE	TRAN
S_DATASET	ACTVT	34 (Write)
	FILENAME	*
	PROGRAM	SAPLSABC SAPLSTRF
S_ADMI_FCD	S_ADMI_FCD	SCMD (Write)

2.3.2.3 Creating a Subaccount in the Global Account

A subaccount enables you to run applications and services in the global account of your ABAP environment.

Prerequisites

You need SAP administrator permission to create subaccounts.

Context

You want to establish an RFC connection to your on-premise system to be checked. To do this, you need to create a subaccount for the Cloud Foundry environment.

> **Note**
>
> If you use the `Prepare an Account for ABAP Development` booster, you can skip this step. The booster will perform this task.

Procedure

1. As an SAP administrator, select the *Global Accounts* navigation pane and open the relevant global account in the *SAP BTP Cockpit*.
2. Select the *Subaccounts* navigation pane.
3. Choose the *New Subaccount* button from the header toolbar to create a new subaccount in the Cloud Foundry environment.
4. In the dialog, enter the following data:

a. Define and enter a *Display Name*.
b. [Optional:] Provide a *Description*.
c. Make sure that the *Neo* checkbox is unmarked.
d. Choose the *Provider* and *Region* of your system connection.
e. Define and enter the *Subdomain* to use a custom domain for your company.
f. [Optional:] If required, choose the following checkboxes:
   - *Use for production*
   - *Enable beta features*

5. Choose *Create*.

   The subaccount will be created and added to your subaccounts. This will take a few moments.
6. Open the subaccount you just created.
7. In the *Cloud Foundry* tile, choose *Enable Cloud Foundry*.
8. Choose *Create*.

Results

The new subaccount for the `Cloud Foundry` environment appears in the global account page and displays the details.

This subaccount enables you to connect your cloud connector which contains the RFC connection(s) to your on-premise system(s).

2.3.2.4 Configuring and Administrating the Cloud Connector

The *Cloud Connector* enables the connection between the SAP BTP ABAP environment and on-premise systems in a secured network.

2.3.2.4.1 Connecting the Cloud Foundry Environment Subaccount to the Cloud Connector

Prerequisites

- You have created a subaccount for the Cloud Foundry environment.
- You have installed the Cloud Connector.

Context

You want to add and connect your subaccount for the Cloud Foundry environment to the Cloud Connector.

For more information, see

- Managing Subaccounts
- Cloud Connector

Procedure

1. Go to your *Cloud Connector* and select the *Connector* navigation pane.
2. Choose the *Add Subaccount* button in the title toolbar.
3. In the *Add Subaccount* wizard, enter the *Region* of your server. Make sure that you choose the exact same region as in your Cloud Foundry subaccount.
4. Enter the ID of the *Subaccount* that you have created for the Cloud Foundry environment in the *SAP BTP Cockpit*. You will find the subaccount ID under *SAP BTP Cockpit* > *your Global Account* > *your Cloud Foundry Subaccount* > *Subaccount Details*.
5. *Cloud Connector*, enter the [Optional:] To label the subaccount in the *Display Name*.
6. Enter the login e-mail address of the *Subaccount User*.
7. Enter the *Password* of the *Subaccount User*.
8. [Optional:] To overwrite the default location ID of the data center in the Cloud Foundry environment, enter the relevant *Location ID*.
9. [Optional:] Provide a *Description* for further details.
10. To add your subaccount to the *Cloud Connector*, choose *Save*.

Results

A new entry for the Cloud Foundry subaccount is added in the *Subaccount Dashboard* table in the Cloud Connector.

2.3.2.4.2 Configuring the Access Control (RFC) for the Remote Connection

The resource accessible represents the on-premise system that can be accessed through the *Cloud Connector*.

Context

You want to enable the connection for SAP BTP ABAP environment to an on-premise system.

For more information, see Configure Access Control (RFC)

Procedure

1. In the *Cloud Connector*, select the *[Display Name of the Cloud Foundry Subaccount]* > *Cloud to On-Premise* navigation pane.
2. To add a new system mapping, choose + from the section toolbar of the *Mapping Virtual To Internal System* table.
3. In the *Add System Mapping* wizard, provide the following data:
   a. Choose *ABAP System* from the *Back-end Type* drop-down list box.
   b. Choose *Next*.
   c. Choose *RFC* as *Protocol* from the drop-down listbox.
   d. Choose *Next*.
4. > **Note**
   > To use a connection **with** load balancing, skip this step and proceed with the next step.

   To use a connection **without** load balancing, proceed as follows:
   a. Choose the *Without load balancing (application server and instance number)* radio button.
   b. Choose *Next*.
   c. Enter the following connection details of the on-premise system:

      > **Note**
      > You will find these details in the SAP Logon of your on-premise system.

      - *Application Server*.
      - *Instance Number* of your Application Server ABAP.
      - Name of the *SAProuter*, if applicable.
   d. Enter a virtual name for the *Application Server*.
   e. Enter a virtual *Instance Number*.
   f. Choose *Next*.
   g. [Optional:] Enter a description.

h. Choose *Next*.

5. > **Note**
   >
   > If you use a connection **without** load balancing, skip this step. Follow the instructions in the previous step instead.

   To use a connection **with** load balancing, proceed as follows:

   a. Choose the *With load balancing (system ID and message server)* radio button.
   b. Choose *Next*.
   c. Enter the following connection details of the on-premise system:

      > **Note**
      >
      > You will find these details in the SAP Logon of your on-premise system.

      - *Message Server*.
      - *System ID*.
      - Name of the *SAProuter*, if applicable.

   d. Choose *Next*.
   e. Define and enter a virtual name for the *Message Server*.
   f. Define and enter a virtual *System ID*.
   g. [Optional:] Enter a description.
   h. Choose *Next*.

6. [Optional:] You can mark the checkbox *Check Internal Host* to immediately check the connection during creation.
7. To trigger creation, choose *Finish*.

   You have added the relevant on-premise system as resource accessible.

   Now, you want to define the permitted function modules as resources for a specific back-end system.

   To do this, proceed as follows:

8. In the *Cloud To On-Premise* navigation pane, select the relevant system mapping in the *Mapping Virtual to Internal System* section.
9. To define which function modules are executable in the on-premise system, choose ↑ *Import resources belonging to a scenario* from the section toolbar in the *Resources Of ...* section.

   > **Note**
   >
   > To get the relevant import file for the Custom Code Migration scenario, see SAP Note 2861842 - Custom Code Migration in SAP BTP ABAP environment: Set up SAP Cloud Connector.

10. *Browse* the *Scenario File* and choose *Select*.

Results

You have defined the on-premise system(s) that are accessible from the ABAP environment.

You can now call the defined function modules in the on-premise system.

2.3.2.5 Administrating User Assignments and Maintaining Communication Arrangements

To assign the `Custom Code Migration` app to the users and to configure the connectivity between the ABAP environment and the Cloud Connector, proceed as follows.

2.3.2.5.1 Administrating User Assignments

The user assignment describes the relevant roles that a user requires to access SAP BTP ABAP environment using the Custom Code Migration Migration app.

Context

You want to enable business users to access the Custom Code Migration app.

For more information see, How to Create a Business Role from a Template.

> **Note**
>
> If you're using the `Prepare an Account for ABAP Development` booster, you can skip this step. The booster will perform this task.

Procedure

1. In the SAP Fiori launchpad, navigate to the *Identity and Access Management* section and open the `Maintain Business Roles` app.
2. On the initial screen, select *Create from Template* from the bottom of the page.

   The *Create Business Role from Template* wizard is opened.
3. Use the value help to choose *SAP_BR_IT_PROJECT_MANAGER* as the underlying *Template*.
4. Enter *BR_IT_PROJECT_MANAGER* as the name of the *New Business Role ID*.
5. Enter a *New Business Role Description*.
6. The checkbox *Create and Assign Spaces Based on SAP-Delivered Spaces* provides you with the following two options:
   a. **Mark** the checkbox to create a new space for the Custom Code Migration app in your subaccount. Define and enter a *New Space ID*.
   b. **Unmark** the checkbox to use an existing space for the app. Consequently, you must allocate the Custom Code Migration app to the existing space and page in your subaccount. For more information, see Working with Predefined Spaces and Pages.
7. Confirm the creation with *OK*.

   The new business role is created and opened in the *Maintain Business Role* page.

8. Select *Maintain Restrictions* from the title toolbar.

   The relevant subpage is opened.
9. In the *Write, Read, Value Help* tab, choose *Unrestricted* from the drop-down list to enable write access for this business role.
10. Navigate back to the main page of your new business role.
11. Open the *Assigned Business Users* tab and add the relevant users using the *Search* field or the *Add* button.
12. *Save* the changes from the bottom.

Results

The business role is created and added. The assigned users now have access to the Custom Code Migration app.

2.3.2.5.2 Maintaining Communication Arrangements

A communication arrangement describes a communication scenario with a remote system during configuration time. It provides the required metadata for the service configuration.

Prerequisites

To create these communication arrangements, you need the `Administrator` role.

You have created a destination to your on-premise system in the *Cloud Connector* that is connected to your BTP subaccount.

Context

You want to enable communication from your ABAP environment to your on-premise systems using Remote Function Calls (RFC).

To do this, you need to use the communication arrangement `SAP_COM_0464` (SAP Custom Code Migration Integration).

Procedure

1. Log on to the *SAP Fiori launchpad* of your ABAP environment.

2. In the *Communication Management* section, select the *Communication Arrangements* tile.
3. Choose *New*.
4. Use the value help to choose the `SAP_COM_0464` scenario.
5. Define and enter an *Arrangement Name*.

   If not yet available or defined, create a new communication system for the communication arrangement to define an endpoint for your checked system.

   a. For the *Communication System*, choose *New*.
   b. In the *New Communication System* dialog, enter the *System ID* and *System Name* of the checked system.
   c. Choose *Create*.
   d. In the *Technical Data* tab under *General*, enter the virtual host as specified in your *Cloud Connector* as *Host Name*. The field *Port* has already been filled in automatically with the default *443*. The value of the field *UI Host* can be changed if it differs from that of the host.
   e. Turn on the slider for *Cloud Connector*.
   f. Filling in the field *SCC Location ID* is optional.
   g. Under *RFC Settings*, fill in the fields *Client*, *Instance Number* and *Target Host* as specified as virtual host in your *Cloud Connector*.
   h. Under *User for Outbound Communication*, choose + to assign the RFC user you created in the checked system to the communication system.
   i. Choose *Create* and then *Save* to save the communication system and to be navigated back to the Communication Arrangement creation page.
6. Under *Additional Properties*, fill out the fields *Object Provider* and *System Group*. This is the name you then select as your *Object Provider to Remote System* for your custom code migration project in the `Custom Code Migration` app.
7. Under *Outbound Services* > *Retrieve Custom Code*, ensure that the *Service Status* is set to *Active*. Optionally, you can test the connection from the SAP BTP ABAP environment system to the checked system by choosing the *Check Connection* button.
8. Under *Outbound Services* > *Display Custom Code*, ensure that the *Service Status* for displaying custom code via UI link is set to *Active*. You may also need to enter a value for *Port* if the system number is not 00.
9. Choose *Save* to save the communication arrangement. A message should now pop up at the bottom of the screen telling you that the activation was successful.

Results

You can now select the communication arrangement in the *Object Provider to Remote System* field in the `Custom Code Migration` app to establish the connection to your on-premise system.

2.3.2.6 Installing Remote Stubs

During ATC execution, the central check system accesses the systems in your landscape remotely through so-called remote stubs using RFC connection. Thus, remote stubs serve as an interface between the central check system and on-premise systems; they return a model from custom code that needs to be checked.

Prerequisites

In the on-premise system, implement SAP Note 2599695 (Custom Code Migration Fiori App: Remote Stubs for the Checked System) in advance.

Context

You want to analyze your custom code in your on-premise system using the `Custom Code Migration` app.

Procedure

1. In the on-premise system, perform transaction `SNOTE`.
2. Download SAP Note 2599695.

   The *Note Assistant: Note Download* dialog is opened.
3. To implement the remote stubs, select and confirm the displayed SAP Note.
4. It is also recommended to apply SAP Notes in checked systems. Update your check system by following these steps:
   a. Perform transaction `SNOTE`.
   b. In the menu bar, select *Goto* > *Other tools* > *Launch Note Analyzer*.
   c. Upload the file *SAPNote_2436688_Checked_System.xml* available as attachment in SAP Note 2436688. You can now process the notes using SAP Note Analyzer.

      Please also apply the following SAP Notes in the **checked system**:

      - 2485231 - Remote ATC Checks of Modifications and Enhancements
      - 2270689 - RFC Extractor for performing static checks
      - 2190065 - ATC/CI: Remote Code Analysis - Object Provider Stub

Results

The `Custom Code Migration` app now has access to the on-premise system. You can now start the custom code analysis.

2.3.2.7 Applying SAP Notes

Apply the following SAP Notes in the **checked system**:

- 2485231 - Remote ATC Checks of Modifications and Enhancements
- 2270689 - RFC Extractor for performing static checks
- 2190065 - ATC/CI: Remote Code Analysis - Object Provider Stub

In addition, the following SAP Notes contain further SAP Notes that are recommended to apply in the checked system:

- 2436688 - Recommended SAP Notes for using SAP S/4HANA custom code checks in ATC
- 2364916 - Recommended SAP Notes for using ATC to perform remote analysis

2.3.3 Custom Code Analysis with SAP Fiori App (Recommended)

The `Custom Code Migration` app enables you to scope and analyze custom code that needs to be migrated from an SAP Business Suite system to SAP S/4HANA. In addition, this app supports you with identifying unused custom code based on your collected usage data. This enables you to remove unused custom code during the system conversion.

Key Features:

Analysis of custom code:

- Graphical representation of custom code analysis results
- Results can be filtered by various categories, such as quick fix availability, scope and usage data.

Scoping:

- Based on usage data, you can define which ABAP custom code needs to be migrated to SAP S/4HANA.
- This app creates a deletion transport in order to delete unused ABAP source code during the system conversion to SAP S/4HANA.

The `Custom Code Migration` app enables you to create an SAP S/4HANA Migration Project.

With the help of an SAP S/4HANA Migration Project, you can analyze custom code that shall be migrated from an existing product like SAP Business Suite to SAP S/4HANA.

> **Note**
>
> In addition to the SAP S/4HANA Migration Project, the `Custom Code Migration` app offers two additional project types.
>
> - The SAP BTP Analysis Project enables you to analyze custom code for readiness to run in SAP BTP ABAP environment.
> - The Custom Code Analysis Project enables you to analyze custom code with arbitrary ATC check variants.
>
> These project types are not relevant for the custom code migration process and will not be described in this guide. To find out more about the SAP BTP Analysis and Custom Code Analysis project, you can use the

> in-app help and the guided tours in the `Custom Code Migration` app and/or consult the Custom Code Analysis SAP Help Portal documentation.

2.3.3.1 Creating a Custom Code Migration Project

A custom code migration project defines the scope of custom code you want to analyze and migrate to SAP S/4HANA.

Procedure

1. Select the tile *Custom Code Migration* from the SAP Fiori launchpad.
2. Choose *Create* to create a new custom code migration project.
3. Enter a *Project Description* for your new project.
4. Define the *Transition Scenario*, the *Target Release* of the back-end and the *Material Number Length*.

   > ⓘ Note
   >
   > Make sure that the checkboxes for *Extended Amount*, *Extended Stock/Requirement Segment* and *Extended Season/Theme/Collection* are enabled, if you intend to use them.

5. In the *System to Be Analyzed* section, define the *Location of Custom Code* and the *Object Provider to Remote System*. You define the object provider in the communication scenario.
6. In the *Advanced Configuration* tab, under *Analysis Configuration*, you can either leave the check box *Start Analysis at Project Creation* checked and start the custom code analysis automatically at project creation or uncheck it and start the analysis manually at a later point.
7. To confirm your input, choose *Create*.

2.3.3.2 Defining the Scope of Your Custom Code Migration Project

The Custom Code Migration app supports you with identifying your unused custom code based on your collected usage data. With the scope of a custom code migration project, you can specify which of your custom code needs to be converted to SAP S/4HANA.

To define the scope of your custom code migration project, you can either use your usage data collected with transaction SUSG or your usage data collected in your Solution Manager (this requires Solution Manager 7.2 support package 8 or higher).

> ⓘ Note
>
> If you have not aggregated your usage data so far, you should activate the aggregation of your usage data in transaction **SUSG** in your production system now. To do so, start transaction **SUSG** in your production system and choose *Activate* to activate the aggregation of your usage data.

> If you have applications that you only use in your development or test system, we recommend that you also collect usage data from these systems.
>
> Ideally, the usage data you add to your custom code migration project should cover at least one year of usage information, so that also usage data of quarter and year ending functionality is considered.

2.3.3.2.1 Adding Usage Data Stored in SUSG

To define the scope of your custom code migration project, you can use your usage data collected with transaction `SUSG`.

Prerequisites

> ⓘ Note
>
> If you have applications that you only use in your development or test system, we recommend that you also add usage data from these systems.

Procedure

1. Start transaction `SUSG` in your production system.

   The *Usage Data: Aggregation State* view is opened.
2. Choose *Create Snapshot* to create a snapshot of your usage data.
3. Recommended: Transfer your usage data from your production system to your checked system.
   a. Choose *Manage Snapshots*.
   b. Choose *Download to File* to export your snapshot as a file.
   c. Choose *Manage Snapshots* in transaction `SUSG` in your checked system.
   d. Choose *Upload from File* to import your usage data from a file.
4. Add the snapshot of your usage data to your custom code migration project in the `Custom Code Migration` app.
   a. Choose *Edit*.
   b. In the *Usage Data* frame, choose *Add*.
   c. Choose ⧉ beneath the *Usage Description* field.
   d. Search for your usage data you created as a snapshot.

   > ⓘ Note
   >
   > If you transferred your usage data to your checked system, your usage data is displayed automatically in the *Items* overview.

e. Select your usage data to add it to your custom code migration project.

> **Note**
>
> When you add usage data to your custom code migration project, a default scope is calculated. The default scope includes all used objects, all objects which are statically referenced by the used objects, and objects for which no usage data is available, such as database tables or data elements.

5. Choose *Scope* to display the scope information.

   Here you can see how many objects are in and out of scope.

> **Tip**
>
> Choose *View By* to display detailed scope information or your scope sorted by object name or object type.

2.3.3.2.2 Adding Usage Data Stored in SAP Solution Manager

You can also upload your usage data collected in the SAP Solution Manager (using UPL or SCMON).

Prerequisites

Your SAP Solution Manager system must be on the release 7.20 (SP8) or higher.

The following SAP Notes must be implemented:

- 2599695
- 2190065

On the central check system, the RFC destination to the SAP Solution Manager system must be created in the transaction SM59. See Configuring the Connection Between Central Check System and Checked System [page 14].

> **Note**
>
> If you have applications that you only use in your development or test system, we recommend that you also add usage data from these systems.

Procedure

1. Add the usage data to your custom code migration project in the `Custom Code Migration` app.
   a. Choose *Edit*.
   b. In the *Usage Data* frame, choose *Add*.

2. Use the value help in the *Destination for Usage* field.
3. Enter the destination to your SAP Solution Manager system in the *Connection* field and choose your destination in the corresponding table row.
4. In the *Usage Description* field, use the value help to select the right SAP Solution Manager usage data snapshot from the list.

Results

The Solution Manager usage data will be added to your custom code migration project.

> **Note**
>
> When loading usage data collected in SAP Solution Manager, the usage counter will not display usage numbers (0 usages). Usage counter data is not supported by the APIs of SAP Solution Manager.
>
> You can find the usage counter here: *Scope* > *Change Scope* > *Entry Points (List Page)* > *Choose a Request Entry Point* > *Procedures within the Entry Point* > *Procedures* > *Number Usages*.

2.3.3.2.3 Changing the Scope of Your Custom Code Migration Project

In the Custom Code Migration app, you can change the scope on the following levels:

- **Request Entry Points**
  Request entry points are all applications or services which are used in your system, such as programs, transactions, URLs, etc. For each request entry point, the used objects are recorded.
  In the initial scope, all used request entry points are added to the scope. You can remove request entry points from the scope by executing the following steps:
  1. Choose *Scope*.
  2. Choose *Change Scope*.
  3. Choose *Entry Points* (list page).
  4. Select the request entry point you want to remove from the scope.
  5. Choose *Remove Entry Point from Scope*.
- **Packages**
  You can add or remove whole packages to/from the scope by executing the following steps:
  1. Choose *Scope*.
  2. Choose *Change Scope*.
  3. Choose *Packages* (list page).
  4. Select the package you want to add to or remove from the scope.
  5. Choose *Add Package to Scope* or *Remove Package from Scope*.
- **Objects**
  You can add or remove single objects to/from the scope by executing the following steps:
  1. Choose *Scope*.

2. Choose *Change Scope*.
3. Choose *Packages* (list page).
4. Navigate to the package of the objects.
5. Select the object you want to add to or remove from the scope either in the list "Scoped Objects" or in the list "Not Scoped Objects".
6. Choose *Add Object to Scope* or *Remove Object from Scope*.

When is an object in scope or out of scope?

Objects are in scope, if ...

- ... they are added to the scope explicitly (by object),
- ... their package is added to scope explicitly,
- ... they are used by at least one request entry point which is in scope,
- ... they are referenced by any other object which is in scope,
- or their object type has no usage data.

Objects are out of scope, if ...

- ... they are removed from scope explicitly (by object),
- or they are not used by any request entry point.

Example

When do you need to change the scope of your custom code migration project?

- **Scenario 1 – Remove transaction from scope**
  A transaction that is used in your Business Suite system shall not be used any longer, because you want to change your business processes in SAP S/4HANA.
  Choose the filter *Scope by Request Entry Points* for your transaction and remove the request entry point from your scope.
- **Scenario 2 – Add new development**
  A new functionality is in development but has not been used in your production system yet. Therefore, no usage data has been collected for this application and it has not been added to your scope automatically.
  Choose the filter *Scope by Package* for all packages containing your developments for the new functionality and add them to your scope.
- **Scenario 3 – Remove complexity**
  Sometimes, your ABAP code can contain complex objects that very often also experience the most frequent changes. With the `Custom Code Migration` app it is possible to identify the most complex objects via complexity analysis and help you decide whether you want to redesign and therefore simplify them.
  In your custom code migration project, choose the bubble chart to display all packages that contain custom code. With this chart, you can also display the Halstead Difficulty of a package. The Halstead complexity measures are software metrics that describe the effort to develop and understand the complexity of a program.

2.3.3.2.4 Removing Unused Custom Code During System Conversion

When you have specified your scope in your custom code migration project, you can create a deletion transport in the checked system. This deletion transport contains all objects which will not be migrated during the system conversion.

Procedure

Choose *Create Deletion Transport* to create the deletion transport.

The deletion transport is also shown in the field *Request/Task* in your custom code migration project.

Next Steps

For more information, see chapter "Integration of Customer Transports" in the corresponding upgrade guide for SAP HANA DB Software Update Manager 2.0.

2.3.3.3 Analyzing the Findings

When you create a custom code migration project, the SAP S/4HANA custom code checks are performed automatically. The Custom Code Migration app then gives you an analytical representation of SAP S/4HANA custom code check findings.

Procedure

1. Choose *Analysis* to get an overview of the results, as soon as the first analysis has been finished.
2. Choose *Analyze Findings* to get a detailed analysis.
3. In the *Custom Code Migration - Analysis - Findings* view, you can specify various filters, for example:
   - *Simplification Item Category*: The simplification item category specifies whether findings are related to functionality which is not available any more or has been changed in an incompatible way.
   - *SAP Note Number*: Lets you analyze findings of a specific Simplification Item (for example field length extension of material number)
   - *Scope Information*: Lets you filter the findings to show only findings for objects that are in scope.
   - *Quick Fix Availability*: Lets you analyze which findings can be solved by a Quick Fix and which findings have to be solved manually.
4. Choose ⚙ to change the settings for the chart.

   To show the Quick Fix availability per SAP Note title, you must specify the following settings for the chart:

- SAP Note title (as Category)
- Findings (as Axis 1)
- Quick Fix Availability (as Series)

5. [Optional:] If you're leveraging AI capabilities in this workflow, please see Working with the Explain Capability In-App.

Results

You can now continue with chapter Custom Code Adaption Before System Conversion [page 62].

> **Note**
>
> It is recommended to focus on Priority 1 and 2 findings first. See also SAP Note 3735015 for adaptions to priorities for SAP S/4HANA custom code checks.

2.3.3.4 Custom Code Adaption Before System Conversion

Before the technical conversion, you have the option to work with transferable quick fixes to ensure that code is adjusted to avoid syntax errors during and after system conversion. Quick fix instructions can be downloaded from the Analyze Custom Code/Custom Code Migration app and can be applied directly on SAP ECC systems before the SAP S/4HANA conversion. Apply SAP Note 3756014 to enable the functionality in your system.

Procedure

Download quick fixes as a ZIP file and upload them in your checked system. You can do this via transaction `ATC` in SAP GUI on your SAP ECC system:

1. In the Analyze Custom Code/Custom Code Migration app, in your project, select the *Analysis* tab.
2. Now select *Analyze Findings*.
3. Select the arrow next to the export icon and choose *Export Transferable Quick Fixes*. The ZIP file is downloaded.
4. In SAP GUI, in transaction `ATC`, select *Upload and Run ATC Quick Fixes*.
5. Select *Upload Quick Fixes* and choose the ZIP file you downloaded earlier in the Analyze Custom Code/Custom Code Migration app.

   > **Note**
   >
   > All actions are to be completed from top to bottom. After compelting an action, there will be a green tick next to the button.

6. Select *Set Packages for Data Elements* and specify packages.

7. [Optional:] For objects that aren't local and when you want to create new data elements, specify a transport. Be aware that running quick fixes (see the next step) is only possible when the objects within a transport aren't locked.
8. Select *Run Quick Fixes*. The *Quick Fix Application Log* opens. The quick fixes have been applied.
9. Select *Activate* to activate your changes. In the window *Implement ATC Quick Fixes* that pops up, you will see a list of inactive objects you can now activate.

Results

You can now continue with chapter Custom Code Adaptation After System Conversion [page 64].

3 Custom Code Adaptation After System Conversion

After Software Update Manager (SUM) has performed the technical conversion, you can start adapting your custom code.

3.1 Running Transactions SPDD, SPAU, and SPAU_ENH

SAP provides the adjustment tools SPDD, SPAU, and SPAU_ENH, which enable you to reimplement any modifications related to ABAP Dictionary objects and development objects (such as programs, function modules, screens, interfaces, and documentation) in system upgrades.

To adapt these modifications, see https://help.sap.com/viewer/p/SAP_S4HANA_ON-PREMISE > *Product Assistance* > *English* > *Enterprise Technology* > *ABAP Platform* > *Developing on the ABAP Platform* > *Development Concepts and Tools* > *Application Development on AS ABAP* > *Customer-Specific ABAP Development* > *Changing the SAP Standard (BC)* > *Installing Upgrades, Support Packages, and Enhancement Packages* and SAP Note 2168190 for more information.

> **Note**
>
> Transaction SPDD is performed during the system conversion whereas all other adaptations take place after the conversion.

3.2 Running Local SAP S/4HANA Checks

To check for ATC findings in the converted system, you need to configure a local ATC run series that checks the requested development objects.

3.2.1 Importing the Simplification Database

To detect custom code which needs to be adapted to SAP S/4HANA, SAP provides tools based on the Simplification Database.

Procedure

Import the Simplification Database. For more information, see Importing the Simplification Database [page 18].

3.2.2 Configuring Local ATC Run Series

Prerequisites

Apply all the relevant SAP Notes that are mentioned in 2436688

Procedure

1. Log on to your converted system.
2. Run transaction ATC.

   The *ABAP Test Cockpit Overview* screen appears.
3. In the navigation pane, expand the nodes *ATC Administration* > *Runs* and double-click the *Schedule Runs* entry.

   The screen that appears displays a list of existing run series.
4. Choose the *Create* button in the toolbar.
5. Enter a name for the new series and confirm.
6. Specify the following entries for the new run series:

Field	Description
`<Description>`	Enter a short description for the ATC run series you want to configure. **→ Tip** You can use the template provided by the *Description* field. When you execute the run series, the built-in variables are filled with data for the system, the day of the week, the calendar week, and the year. However, you can also add further texts to these variables, rearrange them, or replace the built-in variables with your own text.

`<Check Variant>`	Enter one of the following check variants:
	
	`S4HANA_READINESS_2025_NO_FLE`
	
	Predefined settings:
	• Material number length = **18**
	• Extended amount field length is **disabled**
	• Extended stock requirement field length is **disabled**
	• Extended season/theme/collection field length is **disabled**
	
	`S4HANA_READINESS_2025`
	
	Predefined settings:
	• Material number length = **40**
	• Extended amount field length is **enabled**
	• Extended stock requirement field length is **enabled**
	• Extended season/theme/collection field length is **enabled**
	
	ⓘ Note
	These product-specific global check variants check all the simplifications relevant for SAP S/4HANA 2025, including those for previous releases.

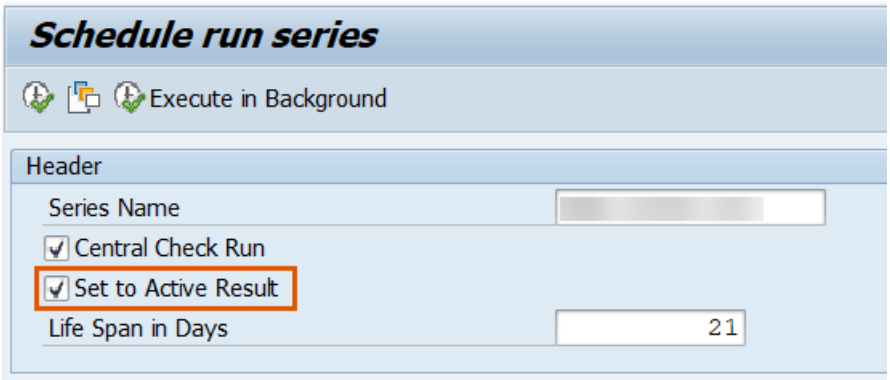
Field	Description
`<Objects to Check>`	Specify the set of development objects you want to check. On the *Checkable Namespaces* tab, the option *By Query* lets you specify objects by name (such as by package name), by transport layer, or by component. The option *By Object Set* lets you specify an object set that you have defined in Code Inspector in the checked system. → Tip You can use the value help to choose the packages or the object set in the remote system. On the *Modified Objects* tab, you can specify the modified source code objects you want to check.

7. Save the configuration.

3.2.3 Scheduling Local ATC Run Series

Procedure

1. In the list of run series, select the run series in question and click the *Schedule* button in the toolbar.
2. In the dialog that appears, choose *Set to Active Result*.



3. Adapt the settings for the *Life Span* of the series and for *Execution* to your needs. Otherwise, leave these settings unchanged.

> ⓘ Note
>
> The *Life Span* defines how long (in days) the ATC result is kept in the system. After this, the ATC result is automatically deleted in the system.

4. Choose *Execute in Background* (F9) to schedule the run series in regular time intervals.

> → Recommendation
>
> SAP recommends to schedule the local ATC run series in regular time intervals since new ATC findings could emerge during custom code adaptation.

3.3 Adapting Custom Code in ABAP Development Tools for Eclipse

After a local ATC run series has checked the requested development objects, you can start adapting the custom code by fixing the ATC findings of the ATC run result with ABAP development tools for Eclipse.

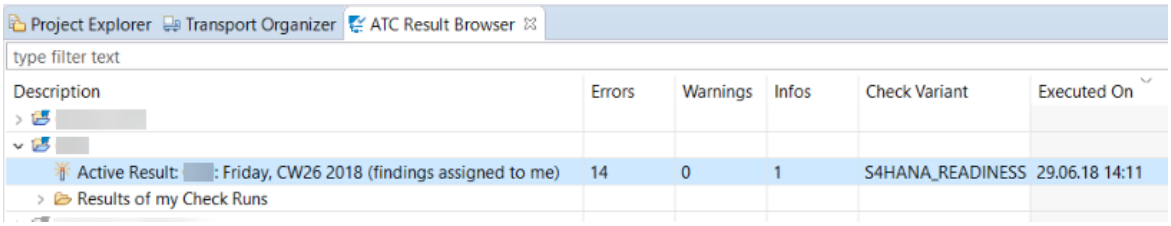
3.3.1 Displaying Active Results

Prerequisites

You have installed the latest version of ABAP development tools for Eclipse. In Eclipse, you created an *ABAP Project* which is connected to your converted SAP S/4HANA system. For more information, see the *Help Contents* in ABAP development tools for Eclipse.

Procedure

1. In ABAP development tools for Eclipse, log on to your converted system.
2. Open the *ATC Result Browser* view and select your converted system.

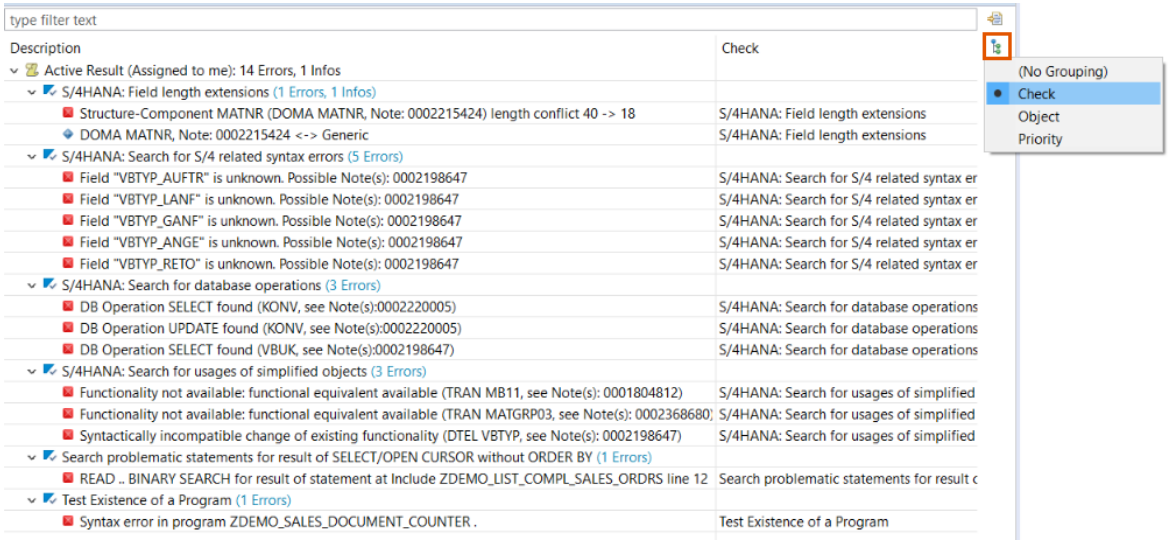


3. Select the *Active Result* to display the list of ATC findings.

> ⓘ Note
>
> By default, all ATC findings are filtered by your user name and only ATC findings belonging to your user are displayed. To display the ATC findings of all users, open the context menu of the *ABAP Project* in the *ATC Result Browser* view, choose *Change User Filter*, and enter * as the user name.

4. Choose *Group By* > *Check* to display the ATC findings sorted by the different checks of the check variant `S4HANA_READINESS`.

> **Note**
>
> Note that you can also use check variant `S4HANA_READINESS_NO_FLE` if you want to exclude the Field Length Extension checks.



> **Tip**
>
> Choose *Configure Columns* to add or remove the columns you want to display or hide in the details of the result. Subsequently, you can filter the active result according to all values in the visible columns.

5. Select the check group *S/4HANA: Search for S/4 related syntax errors* and choose *Recheck* in the context menu.

3.3.2 Optional: Changing the Contact Person of ATC Findings

If you want to assign ATC findings of an ATC run result to a certain developer, you can do that by changing the contact person of the ATC findings.

Prerequisites

You need the authorization object `S_Q_GOVERN` (`ACTVT` = 03 and `ATC_OTYPGO` = 02) to change the contact person.

In addition, you need one of the following authorization objects:

- `S_Q_GOVERN` (`ACTVT` = 01 and `ATC_OTYPGO` = 01) or
- `S_DEVELOP` (`ACTVT` = 02)

> **Note**
>
> You can change the contact person only for local check runs.

Procedure

1. In the *ATC Result Browser* view, select the *Active Result* to display the list of ATC findings.
2. Select all ATC findings in question and choose *Change Contact Person* in the context menu.
3. In the *Change Contact Person* view in the field *New Contact Person*, enter the user name of the developer to whom you want to assign the ATC findings.

   The ATC findings are now assigned to the determined developer.
4. Select the *Active Result* and choose *Refresh* in the context menu.

3.3.3 Fixing ATC Findings

Procedure

1. Open the *ATC Problems* view.

   The view shows a *Worklist* of ATC findings resulting from the recheck on the check group *S/4HANA: Search for S/4 related syntax errors*.
2. Select an ATC finding to display further information about what and where the statements with problems are and what you can do to fix the ATC finding.
3. Double-click an ATC finding to jump to the source code at the position where the ATC found a statement with a problem.
4. Fix each ATC finding in the *Worklist* as described in the *Details* view and the assigned SAP Note.

   > **Note**
   >
   > You can also fix ATC findings by using Quick Fixes. See Applying Quick Fixes [page 71] for more information.

3.3.3.1 Applying Quick Fixes

You can fix certain ATC findings with Quick Fixes. These Quick Fixes provide functions that enable you to resolve errors and warnings without adapting your source code manually.

Context

In the context of the custom code migration, Quick Fixes are available for the following checks and its corresponding simplification items:

Check	Simplification Item	Referenced Object
S/4HANA: Field length extensions	2215852 - Material Number Field Length Extension: Code Adaptions	for example MATNR
	2610650 - Amount Field Length Extension: Code Adaptations	
S/4HANA: Search for database operations	2198647 - S/4 HANA: Data Model Changes in SD	VBFA, VBUK, VBUP
	2220005 - S/4 HANA: Data Model Changes in Pricing and Condition Technique	KONV
	2431747 - General Ledger: Incompatible changes in S/4HANA compared to classic ERP releases	BSEG
S/4HANA: Search for usages of simplified objects	2198647 - S/4 HANA: Data Model Changes in SD	VBTYP
S/4HANA: Search for S/4 related syntax errors	2198647 - S/4 HANA: Data Model Changes in SD	VBTYP
Search problematic statements for result of SELECT/OPEN CURSOR without ORDER BY		

> **Note**
>
> Please note that not all ATC findings resulting from these checks can be fixed with Quick Fixes.

Procedure

> **Note**
>
> ATC findings that can be fixed with a Quick Fix are displayed with a lightbulb icon .

1. Select an ATC finding with a lightbulb icon in the *ATC Problems* view.

2. Open the context menu from the ATC finding and choose *Quick Fix* (`Ctrl` + `1`).

   The *Quick Fix* view opens.
3. Select the displayed Quick Fix and choose *Finish*.

> → Recommendation
>
> If there is more than one Quick Fix available for an ATC finding, we recommend that you select the first Quick Fix displayed.
>
> > ⓘ Note
> >
> > If you're leveraging AI capabilities in this workflow, please see Generating Code Proposals for ATC Findings from SAP S/4HANA Custom Code Checks.

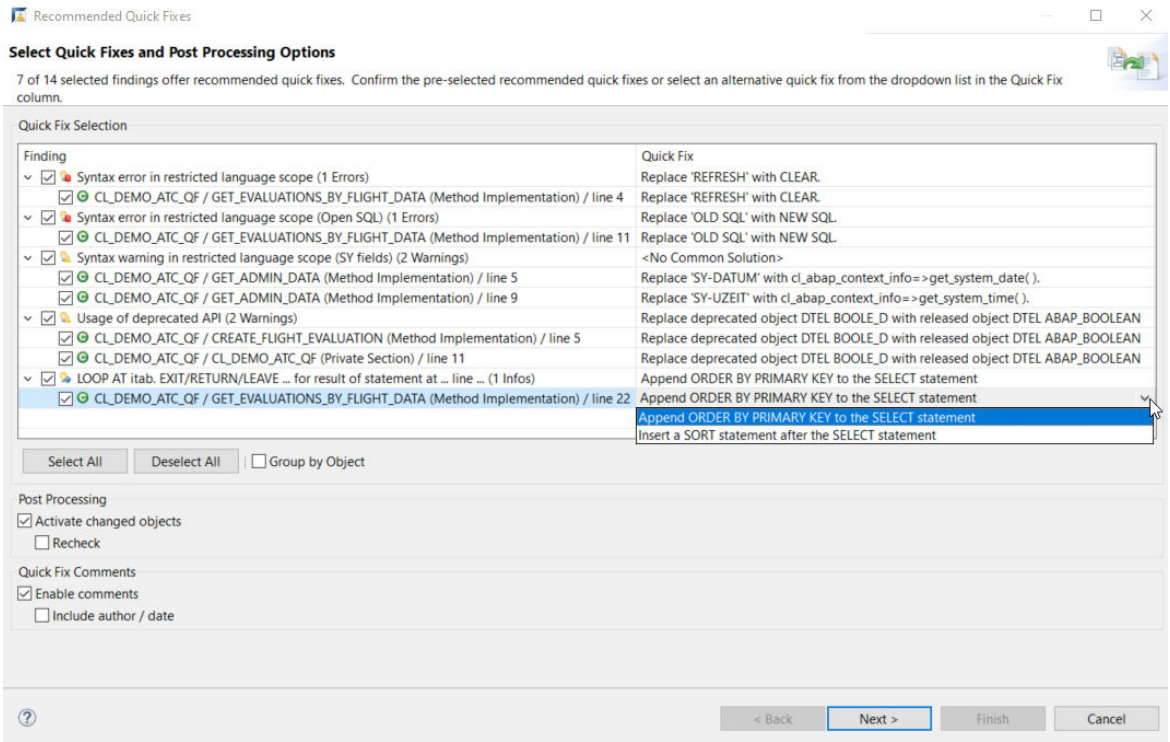
3.3.3.1.1 Applying Recommended Quick Fixes for Multiple ATC Findings

You can also fix multiple ATC findings at once with the *Recommended Quick Fixes* wizard.

Procedure

1. Select multiple ATC findings in the *ATC Problems* view.
2. Open the context menu and choose *Recommended Quick Fixes...*.

   The *Recommended Quick Fixes* wizard is opened.



3. In the *Finding* column in the *Quick Fix Selection* frame, select the ATC findings and the affected objects. By default, all ATC findings and their affected objects are selected.

> → Tip
>
> If you want to display all affected objects and their respective ATC findings, choose *Group by Object*.

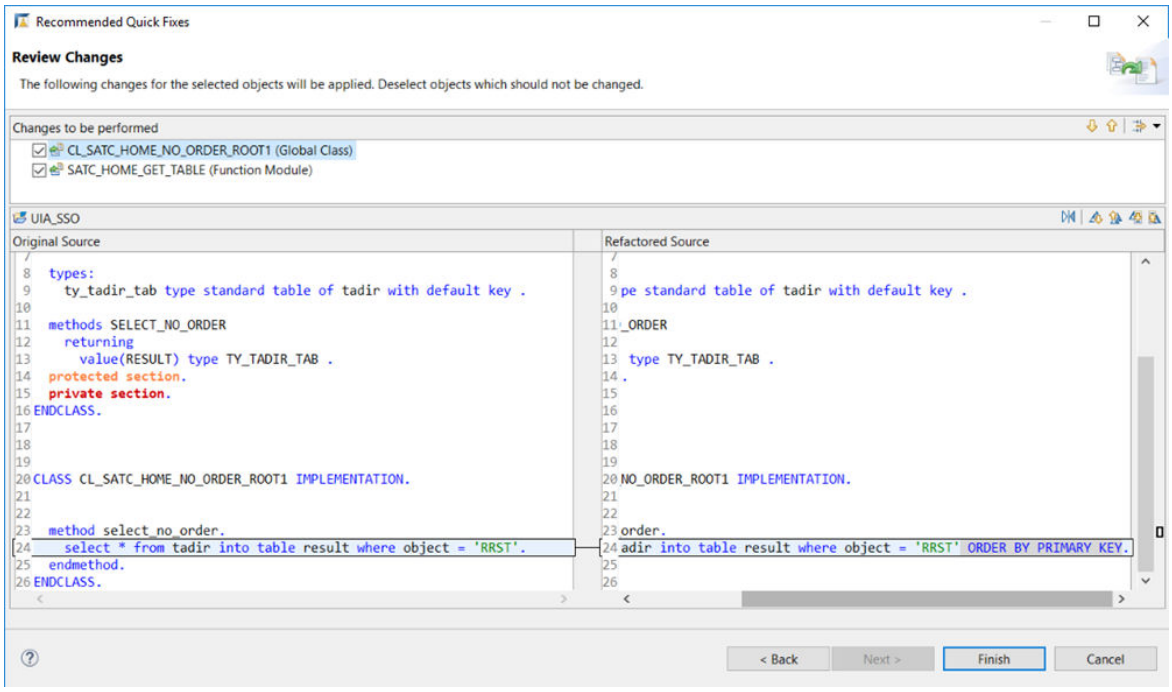
4. In the *Quick Fix* column in the *Quick Fix Selection* frame, the recommended Quick Fixes for the ATC findings are displayed by default. Select a Quick Fix to open a dropdown list with alternative Quick Fixes.

> → Recommendation
>
> We recommend applying the Quick Fixes displayed by default.

5. In the *Post Processing* frame, you can specify that the changed objects are activated after you apply the Quick Fixes. If this option is selected, you can specify that the selected ATC findings are rechecked after you finish the wizard.

> ⓘ Note
>
> If you do not select any post processing options, the initial ATC result is displayed after you finish the wizard. In this case, you have to activate and recheck the ATC findings manually.

6. Choose *Next*.
7. Select a transport request if required.
8. Choose *Next*.
9. Review the changes. Here, a comparison editor is displayed where you can review the refactored source code. The code line where the source code has been refactored is highlighted.



In this example, an `order by primary key` statement was added to the source code to fix the ATC finding.

10. Choose *Finish*.

Results

You applied Quick Fixes for multiple ATC findings at once.

3.3.3.1.2 Adding Quick Fix Comments

This feature enables you to add automatically generated comments to your refactored code.

Context

> **Note**
>
> - In order to keep your code readable and clean, SAP recommends to leave this function deactivated.
> - Quick fix comments are currently available only for transportable objects. It is not possible to apply them for local objects.

Procedure

1. To enable quick fix comments in your system, mark the checkbox *Enable Comments* in the *Quick Fix Comments* section of the *Recommended Quick Fixes* wizard.
2. To include information about the author of the Quick Fix and and the date when it was applied, mark the checkbox *Include author / date*.

   The source code comment in the refactored code contains the following information:

   - Textual description of the applied Quick Fix
   - Your old source code before the Quick Fix
   - ID of related transport request
   - Description from the related transport request
   - Author and date of the applied Quick Fix (if selected)

Example

> **Sample Code**

```
* Quick Fix Delete 'COMPUTE'.
                    * 25.11.2021 18:04:33 AUTHOR_NAME
                    * Transport YKKK012712 Downport Quickfix comments
                    * Replaced Code:
                    * COMPUTE: a = 2 * 4,
```

```
                    a = 2 * 4.
                    * End of Quick Fix
```

This is an example of a Quick Fix comment that has been automatically generated and added to the refactored code. It contains the following information:

- 1. row: applied Quick Fix.
- 2. row: date, time and author name
- 3. row: number and description of the transport request
- 4. row: replaced code as comment
- 5. row: refactored code
- 7. row: end of Quick Fix and Quick Fix comment

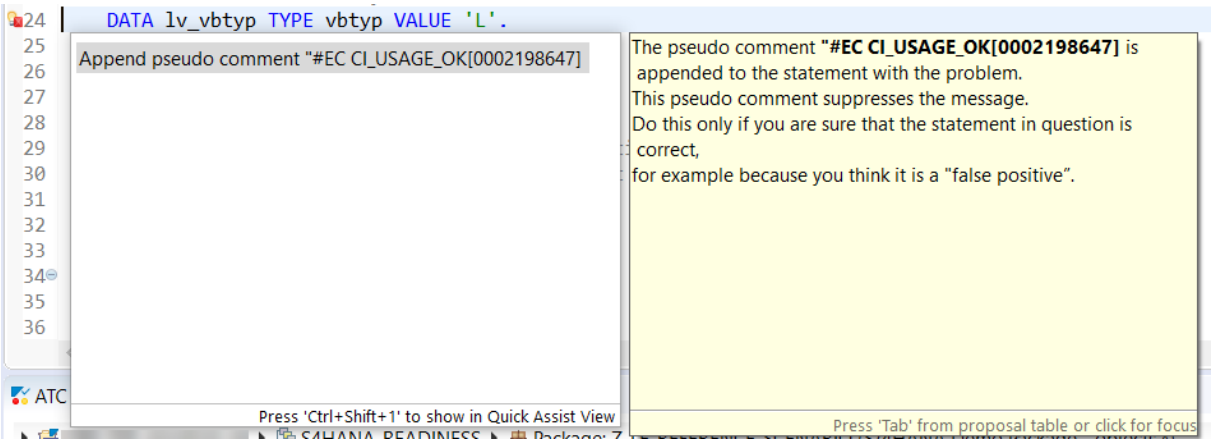
3.3.3.2 Using Pseudo Comments

Pseudo comments are one way of suppressing ATC findings. Sometimes it is the case that certain ATC findings cannot be fixed (so-called *false positives*). In this case and especially in the context of the Custom Code Adaptation to SAP S/4HANA, we recommend that you use pseudo comments to suppress the specific ATC finding.

Procedure

> **Note**
>
> If SAP S/4HANA-related pseudo comments are available for an ATC finding, they can easily be applied as a Quick Fix.

Apply the Quick Fix. For more information, see Applying Quick Fixes [page 71].



The pseudo comment is appended at the end of the code line in question.

> → Remember
>
> The pseudo comment appended does not fix the ATC finding. It only suppresses the ATC finding, so it does not appear anymore after a recheck.

3.3.3.3 Using Exemptions

If you can't clear an ATC finding by correcting the underlying problem, you can still clear it by requesting an exemption. Once approved, an exemption masks an ATC error or warning message. The finding then does no longer appear as an open issue in the ATC results.

> ⓘ Note
>
> The exemption does not fix the ATC finding. It only suppresses the ATC finding, so it does not appear anymore after a recheck.

Using Exemptions in SAP S/4HANA

For more information about creating exemptions and maintinaing approver responsibilities in an on-premise environment, see:

- ATC Exemptions
- Working with ATC Exemptions

Using Exemptions in SAP BTP ABAP environment

In the ABAP environment, exemptions are transportable objects. You can use the ATC exemption editor to edit the details of an exemption, retrieve the ABAP development tools for Eclipse link to trigger an exemption request or, as an approver, make your assessment on an exemption.

For more information, see: Working with ATC Exemptions.

You can also use the Approve ATC Exemptions Fiori app to manage exemptions. SeeApprove ATC Exemptions.

> → Remember
>
> The Approve ATC Exemptions app only supports exemptions when the SAP BTP ABAP environment acts as the central check system and when exemptions are requested via the ATC developer scenario (see Configuring the ATC Developer Scenario). Exemptions of type `CHKE` are not supported. To learn more about how to approve or reject `CHKE` exemptions in ABAP development tools for Eclipse, see Approving and Rejecting ATC Exemptions.

3.3.4 Rechecking the Active Result

Once you have fixed the ATC findings of the check group *S/4HANA: Search for S/4 related syntax errors*, you need to recheck the active result and fix all other ATC findings.

Procedure

1. Open the *ATC Result Browser* view.
2. Right-click *Active Result* and choose *Recheck*.
3. Open the *ATC Problems* view.

   The view shows a *Worklist* of ATC findings resulting from the recheck on the active result.
4. Fix the ATC findings in the *Worklist*.
5. Recheck the *Active Result* in *ATC Result Browser* view again to validate that the ATC findings are fixed.

   > ⓘ Note
   >
   > Repeat the steps 4 and 5 if there are still ATC findings in the *Worklist* after the recheck.

3.4 Rerunning Local SAP S/4HANA Checks

Rerun the local ATC run series once all the ATC findings are fixed, since new ATC findings could have emerged during the adaptation of your custom code.

4 What's Next?

There are a few more things you can do after the custom code migration:

Testing Applications

Test your applications to check whether your programs run on SAP S/4HANA.

> **Note**
>
> ATC is not able to find all potential issues (for example, dynamic coding is not covered by static code checks).

Regular ATC Checks with `S4HANA_READINESS`

Add the SAP S/4HANA checks with check variant `S4HANA_READINESS` to your regular ATC checks.

Runtime Checks

Run the Runtime Check Monitor in transaction `SRTCM` in the new productive system and activate the checks *Empty table in FOR ALL ENTRIES clause* and *Missing ORDER BY or SORT after SELECT*. Schedule the check runs on a regular basis and correct the additional findings found by the runtime check.

Performance Optimizations

Switch on SQL Monitor in the productive system. Sort the SQL Monitor results by *execution time* and optimize the top 10-20 SQLs in your custom code which affect relevant business processes. After performance optimizations reach the productive system, this process needs to be repeated 2-3 times to achieve the best results.

For more information, see ABAP SQL Monitor Implementation Guide and Best Practices

Optimizing Your Code for SAP S/4HANA

After the successful migration, you can now start to think about optimizing your business processes by using code push down techniques of SAP S/4HANA, like the Core Data Services (CDS) and ABAP-managed database procedures (AMDP) implemented by the native HANA language SQL script.

Enabling In-App Help Content

You can access in-app help from the UIs of an app that provides you context-sensitive help information for UI elements and guided tours.

The in-app help is web-based and needs to be configured in the context of ABAP platform. In addition, an embedded help system (for example, the SAP Help Portal) is required.

> **Note**
>
> In your on-premise system, you need to configure the in-app help, before you can benefit from it.
>
> In the context of ABAP environment, the in-app help is enabled per default.
>
> If you want to customize the help content, you need to implement your own help system.

For further information, see:

- Setup of In-Application Help (Optional)

5 Glossary

The following terms are used within the context of this Custom Code Migration guide:

Term	Definition
(ABAP) Custom Code	Describes the development objects and ABAP source code that has been developed by customers or SAP partners.
ABAP Test Cockpit (ATC)	Tool for performing static and dynamic quality checking of ABAP source code and associated repository objects both remotely and locally.
ATC Finding	Message with supporting information and functionality that alerts a developer to a problem with ABAP code or some other object in ABAP Repository.
Booster	In the context of Cloud Integration Automation "CIAS", a set of guided interactive steps that enable a customer to select, configure and consume technical components on SAP Business Technology Platform to achieve a specific technical goal.
Central Check System	Backend system that provides and performs the SAP S/4HANA custom code checks for one or multiple SAP systems remotely.
Checked System	System with custom code that gets checked remotely by the central check system.
Conversion / Migration	Describes the process of bringing custom code from an existing Enterprise Resource Planning (ERP) system to an SAP S/4HANA system.
Simplification Database	Database table in the SAP S/4HANA system that contains all the simplification items referring to SAP objects simplified in SAP S/4HANA.
Simplification Item	Describes the adaptation of an SAP product that has a technical impact on custom code.
Transformation	Describes the process for bringing custom code from an existing on-premise Enterprise Resource Planning (ERP) system to SAP BTP ABAP environment.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

www.sap.com/contactsap

© 2026 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.