Custom Code Migration Guide for SAP S/4HANA 1809

Feature Package Stack 02

THE BEST RUN **SAP**

Content

1 Custom Code Migration Guide for SAP S/4HANA 1809

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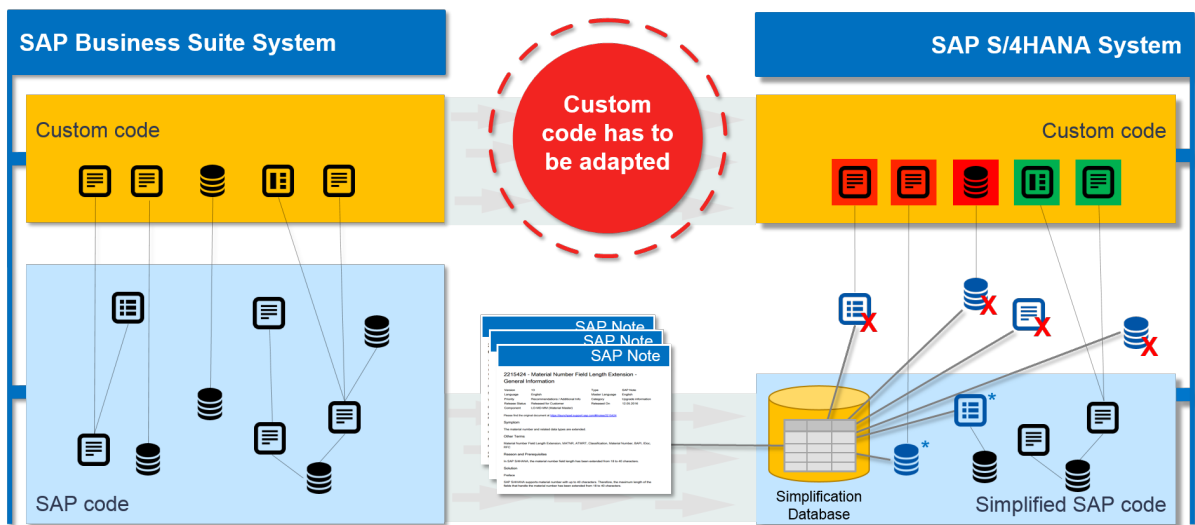
Some of your custom code objects are not valid anymore and either do not perform as expected or produce syntax errors or dumps (red objects in the picture). You almost certainly have other objects that do perform as expected and do not need to be changed (green objects in the picture).

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.

2 Getting Started

This guide focuses on the custom code related process, from checking custom code remotely, then analyzing the findings, and finally adapting the custom code. Read this guide carefully to get instructions on how to adapt your custom code to SAP S/4HANA 1809.

2.1 Overview of the Conversion Process

SAP provides a process for the system conversion to SAP S/4HANA. The following figure gives you an overview of the tools, the phases, and the activities involved in the process.
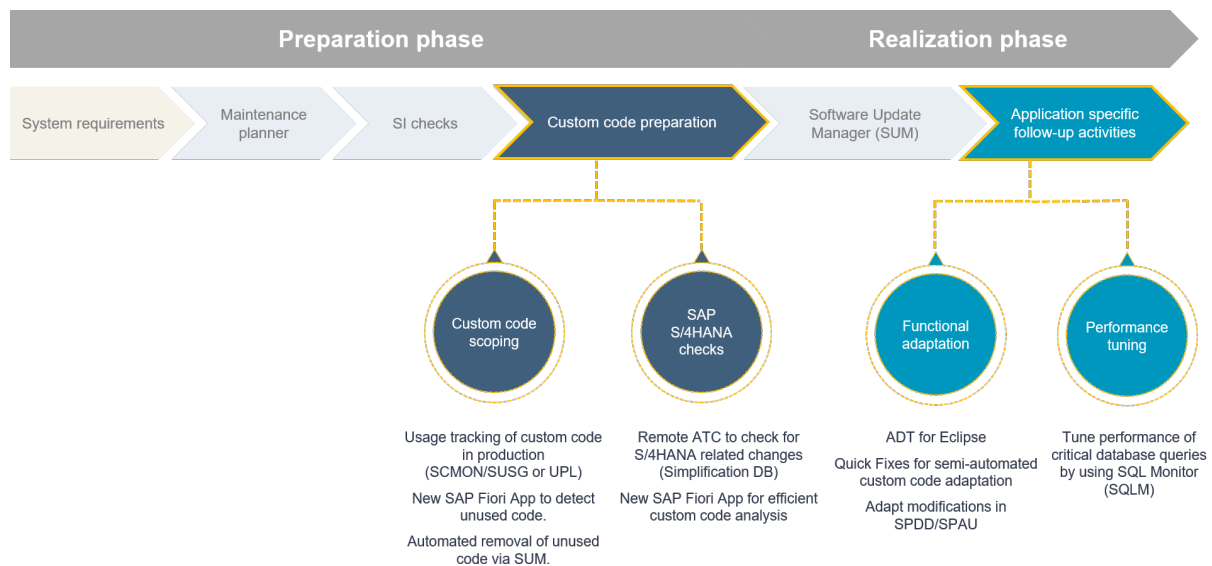
The Custom Code Migration process describes the tools and necessary activities that help you to migrate custom code. The process consists of preparatory analysis (*custom code analysis*) and the adaptation of the custom code (*custom code adaptation*) after the technical conversion.

Custom Code Analysis

To prepare the system conversion, we recommend that you evaluate your custom code and remove any obsolete code as indicated from your aggregated usage data (SUSG) and in the Usage Procedure Log (UPL/SCMON). For more information, see Aggregate usage data in your production system with SUSG transaction🔨 and ABAP Call Monitor (SCMON) – Analyze usage of your code🔨.

In addition, your custom code needs to be checked for any SAP S/4HANA and SAP HANA related changes. You can either do the custom code analysis with the *Custom Code Migration* app or in SAP GUI.

> → Recommendation
>
> We recommend to do the custom code analysis with the Custom Code Migration app, since it provides more functionality and does the analysis automatically.

In SAP GUI, your custom code needs to be checked with ABAP Test Cockpit (ATC) against the SAP S/4HANA simplifications in the Simplification Database. 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 step you can estimate the effort required to adapt custom code to migrate to SAP S/4HANA.

> **i Note**
>
> The only purpose of the custom code analysis phase is to estimate the effort required for the custom code adaptation for system conversion in your current SAP Business Suite landscape. This phase can be a long time before the actual system conversion to SAP S/4HANA.
>
> Nevertheless, in this phase you can also prepare your custom code for the future system conversion. For more information, see the blog What you can do today to prepare your custom code for SAP S/4HANA.
>
> Only after the system conversion to SAP S/4HANA was fulfilled can the functional adaptation be carried out locally on the SAP S/4HANA system.

Custom Code Adaptation

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 (ADT)*. The result is a list of ATC findings that are related to SAP S/4HANA simplifications and refer to SAP Notes which describe how to solve the issues. Based on these ATC findings, you can start adapting your custom code.

For more information on the overall system conversion process, see https://help.sap.com/viewer/p/SAP_S4HANA_ON-PREMISE ▶ *Product Documentation* ❯ *Conversion Guide* ◗.

> **i Note**
>
> If you upgrade from lower release of SAP S/4HANA to SAP S/4HANA 1809, you also need to adapt some of your custom code. For more information on the upgrade process, see https://help.sap.com/viewer/p/SAP_S4HANA_ON-PREMISE ▶ *Product Documentation* ❯ *Upgrade Guide* ◗.

2.2 System Requirements

There are several system requirements for the custom code analysis and the custom code adaptation:

System Requirements for the Custom Code Analysis with SAP Fiori App

To perform the custom code analysis with the Custom Code Migration app, you must set up a SAP S/4HANA 1809 system.

> **i Note**
>
> If you have already setup an SAP S/4HANA sandbox system, this system can also be used for the custom code analysis.

SAP System	Software Component
Central Check System	SAP_BASIS 7.53
Checked System	SAP_BASIS 7.00, 7.01, 7.02, 7.31, 7.40, 7.50, 7.51 or 7.52

System Requirements for the Custom Code Analysis in SAP GUI

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_BASIS 7.00, 7.01, 7.02, 7.31, 7.40, 7.50, 7.51 or 7.52

System Requirements for the Custom Code Adaptation in ADT

To adapt your custom code, you need the ABAP Development Tools (ADT) in the latest client version. See
https://tools.hana.ondemand.com/ for more information.

> **→ Recommendation**
>
> We recommend that you use the ABAP Development Tools (ADT) since transaction `SE80` in SAP GUI no
> longer supports all development objects (such as CDS Views) needed in SAP S/4HANA.

3 Custom Code Analysis During Preparation Phase

The custom code analysis is performed before the technical conversion and is **not mandatory but recommended**. In this phase, you can estimate the effort required to adapt the custom code to SAP S/4HANA simplifications.

> i Note
>
> The functional adaptation of custom code is done after the technical conversion. If you do not want to estimate the effort required in advance, continue with the Custom Code Adaptation After System Conversion [page 27].

You can do the custom code analysis in two ways: Either with the Custom Code Migration app or in SAP GUI.

> → Recommendation
>
> We recommend doing the custom code analysis with the Custom Code Migration app, since it provides more functionality and does the analysis automatically.

For both scenarios, you must first execute some steps to prepare the custom code analysis. See the next chapter Preparing the Custom Code Analysis [page 9] for more information.

Afterwards, continue with chapter 3.2 Custom Code Analysis with SAP Fiori App [page 13], if you want to analyze your custom code with the Custom Code Migration app.

Otherwise, continue with chapter 3.3 Custom Code Analysis in SAP GUI [page 19] to analyze your custom code in SAP GUI.

3.1 Preparing the Custom Code Analysis

The following steps are relevant for the custom code analysis with the Custom Code Migration app as well as in SAP GUI.

3.1.1 Configuring the User

In the **checked system**, the RFC user needs the following authorizations:

Name of Authorization Object	Activity	Type of RFC Object	Name of RFC Object	Description
S_RFC	16 (Execute)	FUGR	SABP_COMP_PROCS _E, SCA_REMOTE_DATA _ACCESS	Authorization for: - Configuring Object Providers - Running SAP S/4HANA checks
		FUNC	REPOSITORY_ENVI RONMENT_ALL, RFC_GET_NAMETAB, FUNCTION_EXISTS, RFCPING	
S_DEVELOP	03 (Display)	*	*	

In the **Central Check System**, you need the following user 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	

3.1.2 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 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

3.1.3 Configuring RFC Connections

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

> **i** Note
>
> The RFC connection must be usable without a logon dialog.

3.1.4 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.

3.1.4.1 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 *Software Downloads* and search for `CCMSIDB`.
3. Choose the relevant ZIP file and choose *Download Basket*.

> **i 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.
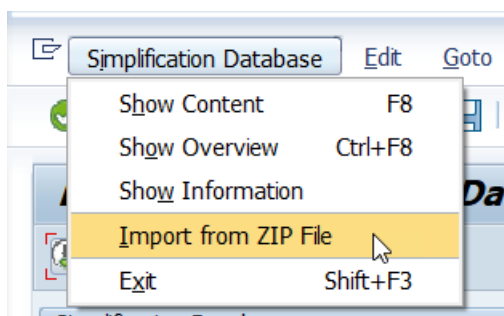
3.1.4.2   Importing the Simplification Database

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.

3.1.4.3 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.

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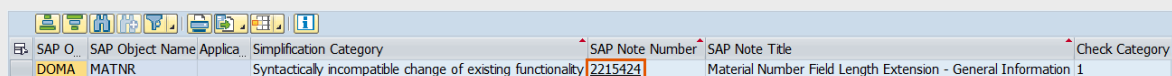
Results

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 |

3.2 Custom Code Analysis with SAP Fiori App

The *Custom Code Migration* app enables you to analyze custom code that needs to be migrated from an SAP Business Suite system to SAP S/4HANA 1809. To evaluate the development objects to be adopted, it performs the SAP S/4HANA custom code checks.
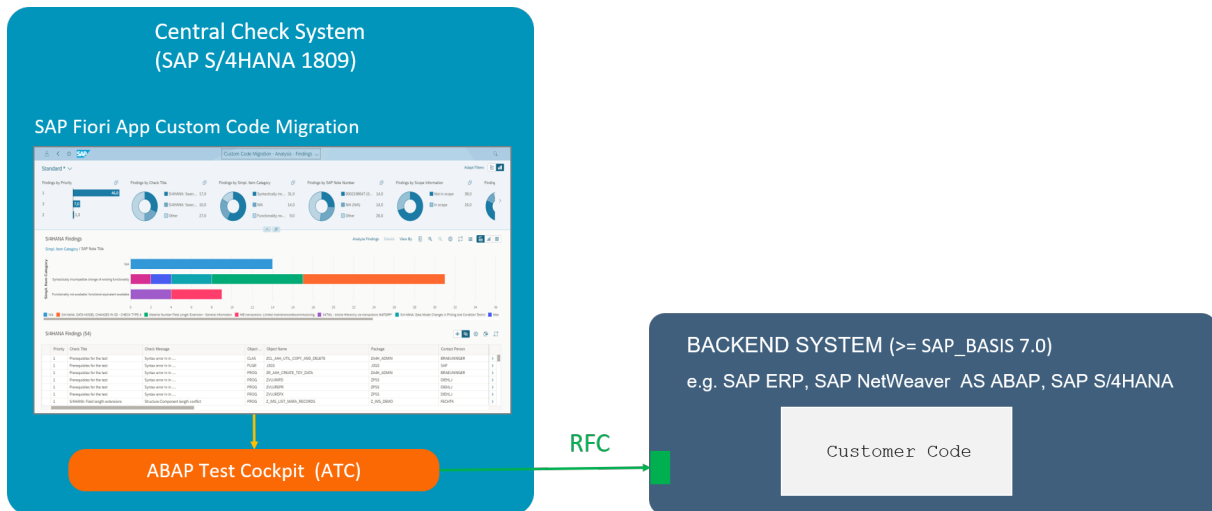
3.2.1 Implementing the Custom Code Migration App

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 General Functions for the Key User.

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

1. In transaction `PFCG`, add the business catalog `SAP_BASIS_TCR_T` (SAP: Application Services) to the user roles that will use the Custom Code Migration app.

   > **i** 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 his 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 by request entry points	`SYCM_APS_C_SCP_BY_EP_CDS`
Custom code scoping by packages	`SYCM_APS_C_SCP_BY_PK_CDS`

For more information about activating OData services, see http://help.sap.com/s4hana `<Your on-premise edition>` *Product Documentation* > *UI Technology Guide* > *Implementation Task on the Front-End Server* > *Activate 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.

Back-End Server: Establishing RFC Connection

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. For this, an RFC destination from the back-end system of the app to the checked system is required.

1. In transaction `SM59`, create an RFC destination to the checked system.

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

3.2.2 Creating a Custom Code Migration Project

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

Procedure

1. Start the tile *Custom Code Migration* from the SAP Fiori launchpad.
2. Choose ✚ to create a new custom code migration project.
3. Enter a description and define the target release and the RFC destination to the SAP system which contains your custom code.

3.2.3 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).

> **i Note**
>
> If you didn't aggregate 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.
>
> 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.

3.2.3.1 Adding Usage Data to your Custom Code Migration Project

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

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 ✛.
   c. Choose ⊡ beneath the *Usage Description* field.
   d. Search for your usage data you created as a snapshot.

   > **i Note**
   >
   > When 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.

   > **i 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.

3.2.3.2 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 like 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 processing the following steps:
  1. Choose *Scope*.
  2. Choose *Scope Request Entry Point*.
  3. Select the request entry point you want to remove from the scope.
  4. Choose *Remove from Scope*.

- **Packages**
  You can add or remove whole packages to/from the scope by processing the following steps:
  1. Choose *Scope*.
  2. Choose *Scope Packages*.
  3. Select the package you want to add to or remove from the scope.
  4. Choose *Add to Scope* or *Remove from Scope*.

- **Objects**
  You can add or remove single objects to/from the scope by processing the following steps:
  1. Choose *Scope*.
  2. Choose *Scope Packages*.
  3. Navigate to the package of the objects.
  4. Select the object you want to add to or remove from the scope either in the list "Not Scoped Objects" or in the list "Scoped Objects".
  5. Choose *Add to Scope* or *Remove 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**
>   A transaction which is used in your Business Suite system shall not be used any longer, because you want to change your business processed 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**
>   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.

3.2.3.3   Removing Unused Custom Code During System Conversion

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

Procedure

Choose *Save Scope as Transport Request* to create the transport request.

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

Next Steps

For more information on the integration of your transports into the system conversion procedure, see chapter 3.21 in the upgrade guide Updating SAP ABAP Systems on UNIX and Linux: SAP HANA DB.

3.2.4 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.

> ⁜ Example
>
> In order 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)
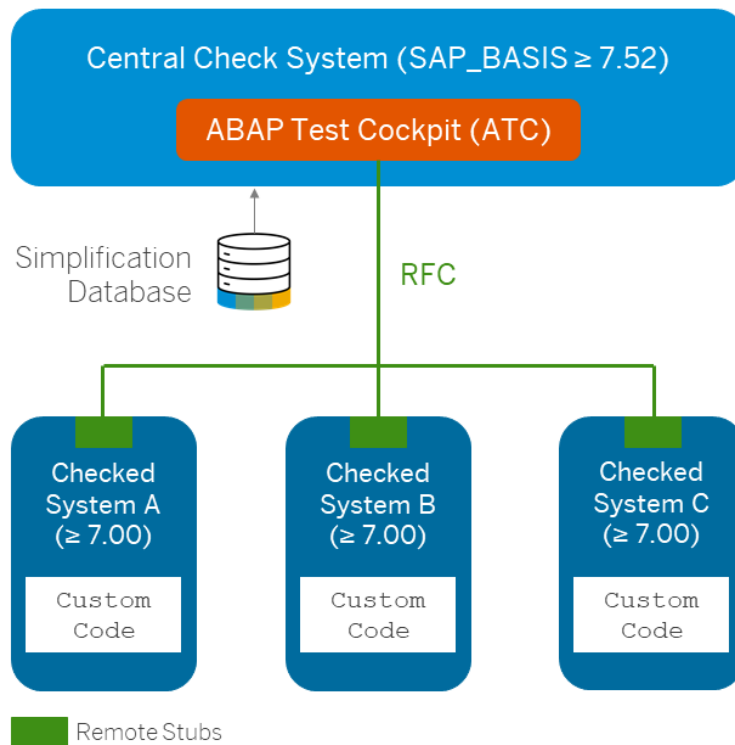> • Quick Fix Availability (as Series)

3.3 Custom Code Analysis in SAP GUI

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.

> i Note
>
> The Central Check System can be used to check one or more SAP Business Suite systems.

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.

Central Check System (SAP_BASIS ≥ 7.52)

ABAP Test Cockpit (ATC)

Simplification Database

RFC

Checked System A (≥ 7.00)

Custom Code

Checked System B (≥ 7.00)

Custom Code

Checked System C (≥ 7.00)

Custom Code

Remote Stubs

3.3.1 Setting the Role as Central Check System

The system role needs to be specified as 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. On the *Change System Role* screen, choose *ATC Checks by Object Providers Only*.

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

3.3.2 Creating System Groups

A system group contains 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. In the navigation pane in transaction `ATC`, expand the node ▶ *ATC Administration* ❯ *Setup* ◀ and double-click *Object Providers*.
2. Double-click the item *System Groups* for selection.
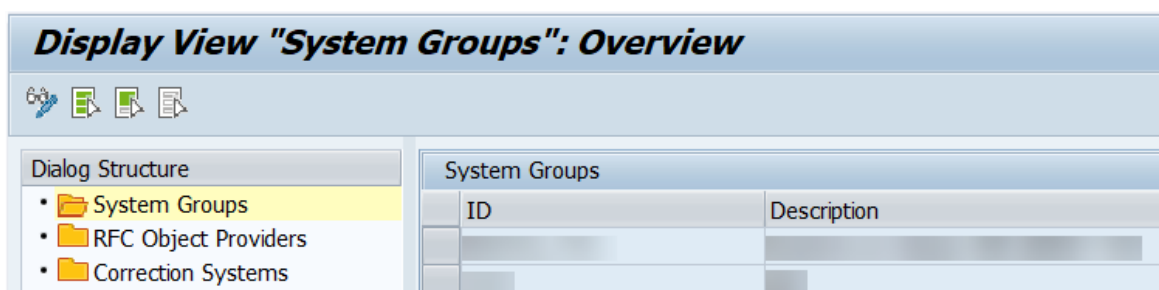
3. Switch to change mode (`F6`).
4. Choose the *New Entries* button in the toolbar.
5. Enter an *ID* and short *Description* for the new system group you want to add.
6. Save the new entry.

3.3.3 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.
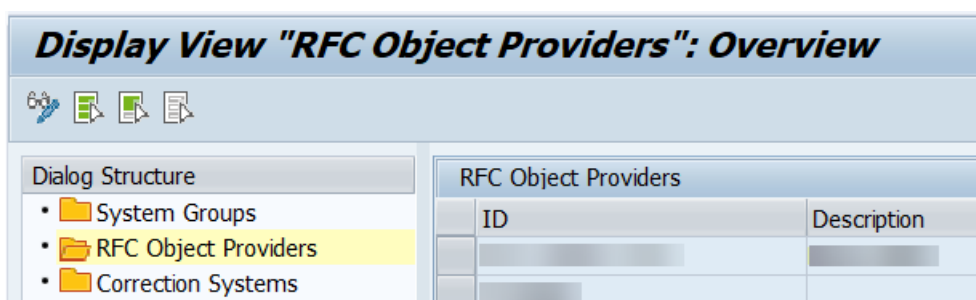
2. Switch to change mode (F6).
3. Choose the *New Entries* button in the toolbar.
4. Specify the following entries for the Object Provider you want to create:

Fields	Description
`<ID>`	ID that specifies the Object Provider i 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 i Note This RFC connection must be usable without a logon dialog.

5. Save the new entry.

3.3.4 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.

3.3.4.1 Configuring a Remote ATC Run Series

Procedure

1. In the navigation pane in transaction `ATC`, expand the nodes ▶ *ATC Administration* ❯ *Runs* ◼ and double-click *Schedule Runs*.

   The screen that appears displays a list of existing run series.
2. Choose the *Create* button in the toolbar.

3. Enter the name for the series you want to create and confirm.

4. 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.
`<Check Variant>`	Enter the check variant `S4HANA_READINESS_1809` i Note This product-specific global check variant checks only the simplifications relating to SAP S/4HANA 1809. → Tip If you do not want to use a material number length of 40 characters in SAP S/4HANA, you can copy the check variant `S4HANA_READINESS_1809` and specify your desired material number length (for example 18 characters) for the check *S/4HANA: Field length extension* in transaction `SCI`.
`<Object Provider>`	Enter the ID of the relevant object provider that represents the remote system you want to check.

Field	Description
`<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.

5. Save the configuration.

3.3.4.2   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.

   > i 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.

3. Choose *Execute* (F8).

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

3.3.4.3 Monitoring an ATC Run Series

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

Procedure

1. In the navigation pane in transaction `ATC`, expand the nodes ▶ *ATC Administration* 〉 *Runs* 〉 and double-click *Monitor and Control Runs*.
2. 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.

3.3.4.4 Analyzing ATC Run Results

Procedure

1. In the navigation pane in transaction `ATC`, expand the nodes ▶ *ATC Administration* 〉 *Runs* 〉 and double-click *Manage Results*.
2. Leave the run series field empty and choose Execute (`F8`).

   > **i 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.
3. 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`.
4. Choose *Statistics View*.
5. Choose *Choose Statistic* to group the ATC findings (for example, by SAP Note Number).
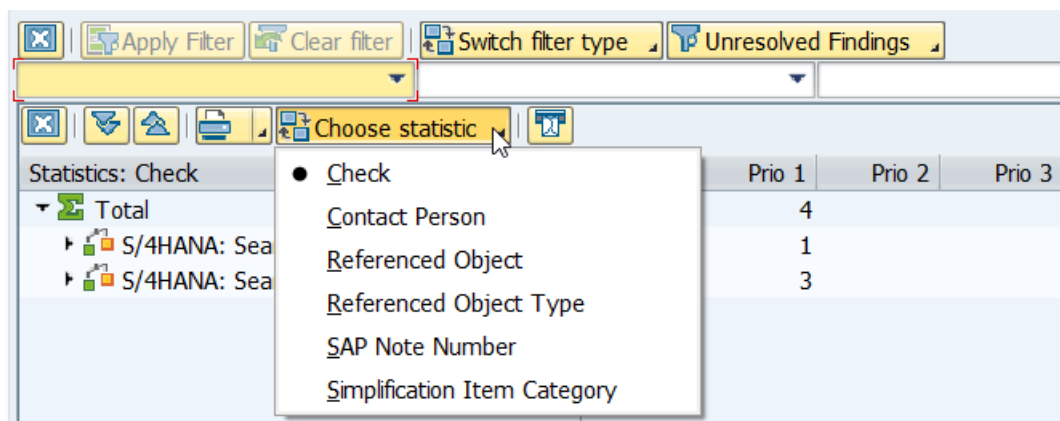
The ATC findings are grouped by the selected statistics.

6. Double-click a statistics group to display the list of ATC findings assigned to the selected statistics.

7. 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

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

4 Custom Code Adaptation After System Conversion

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

> i Note
>
> This phase is separated chronologically from the custom code analysis and can be performed much later (even years) after the first analysis took place.

4.1 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.

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

4.2 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.

4.2.1 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 10].

4.2.2 Configuring Local ATC Run Series

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.
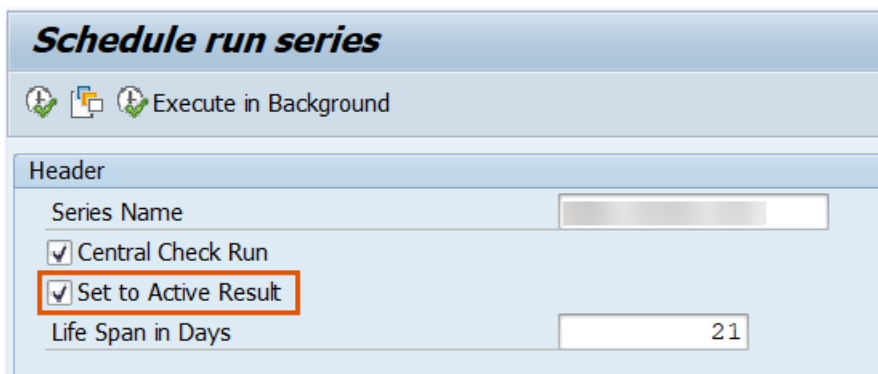6. Specify the following entries for the new run series:

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 the check variant `S4HANA_READINESS`.
	ⓘ Note This check variant provides checks for SAP S/4HANA readiness (for example, searches for DB operations or field length extensions).
	→ Tip If you do not want to use a material number length of 40 characters in SAP S/4HANA, you can copy the check variant `S4HANA_READINESS` and specify your desired material number length (for example 18 characters) for the check *S/4HANA: Field length extension* in transaction `SCI`.
`<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.

4.2.3 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.

> **i** 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
>
> We recommend that youto schedule the local ATC run series in regular time intervals since new ATC findings could emerge during custom code adaptation.

4.3 Adapting Custom Code in ADT

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 (ADT).
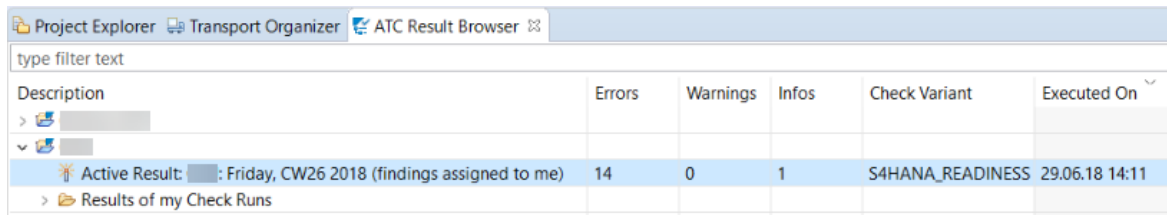
4.3.1 Displaying Active Results

Prerequisites

You installed the latest version of ABAP Development Tools (ADT) in 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 ADT.

Procedure

1. In ADT, 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.

> **i 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.

Description	Check	
✓ 🏛 Active Result (Assigned to me): 14 Errors, 1 Infos		
✓ 🗹 S/4HANA: Field length extensions (1 Errors, 1 Infos)		
◼ Structure-Component MATNR (DOMA MATNR, Note: 0002215424) length conflict 40 -> 18	S/4HANA: Field length extensions	
◆ DOMA MATNR, Note: 0002215424 <-> Generic	S/4HANA: Field length extensions	
✓ 🗹 S/4HANA: Search for S/4 related syntax errors (5 Errors)		
◼ Field "VBTYP_AUFTR" is unknown. Possible Note(s): 0002198647	S/4HANA: Search for S/4 related syntax er	
◼ Field "VBTYP_LANF" is unknown. Possible Note(s): 0002198647	S/4HANA: Search for S/4 related syntax er	
◼ Field "VBTYP_GANF" is unknown. Possible Note(s): 0002198647	S/4HANA: Search for S/4 related syntax er	
◼ Field "VBTYP_ANGE" is unknown. Possible Note(s): 0002198647	S/4HANA: Search for S/4 related syntax er	
◼ Field "VBTYP_RETO" is unknown. Possible Note(s): 0002198647	S/4HANA: Search for S/4 related syntax er	
✓ 🗹 S/4HANA: Search for database operations (3 Errors)		
◼ DB Operation SELECT found (KONV, see Note(s):0002220005)	S/4HANA: Search for database operations	
◼ DB Operation UPDATE found (KONV, see Note(s):0002220005)	S/4HANA: Search for database operations	
◼ DB Operation SELECT found (VBUK, see Note(s):0002198647)	S/4HANA: Search for database operations	
✓ 🗹 S/4HANA: Search for usages of simplified objects (3 Errors)		
◼ Functionality not available: functional equivalent available (TRAN MB11, see Note(s): 0001804812)	S/4HANA: Search for usages of simplified	
◼ Functionality not available: functional equivalent available (TRAN MATGRP03, see Note(s): 0002368680)	S/4HANA: Search for usages of simplified	
◼ Syntactically incompatible change of existing functionality (DTEL VBTYP, see Note(s): 0002198647)	S/4HANA: Search for usages of simplified	
✓ 🗹 Search problematic statements for result of SELECT/OPEN CURSOR without ORDER BY (1 Errors)		
◼ READ .. BINARY SEARCH for result of statement at Include ZDEMO_LIST_COMPL_SALES_ORDRS line 12	Search problematic statements for result o	
✓ 🗹 Test Existence of a Program (1 Errors)		
◼ Syntax error in program ZDEMO_SALES_DOCUMENT_COUNTER .	Test Existence of a Program	

Grouping options shown: (No Grouping), Check, Object, Priority

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

4.3.2 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)

> **i 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.

4.3.3 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.

> **i Note**
>
> You can also fix ATC findings by using Quick Fixes. See Applying Quick Fixes [page 33] for more information.

4.3.3.1 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	

Check	Simplification Item	Referenced Object
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		

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

Procedure

> **i 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. Right-click the ATC finding and choose *Quick Fix* (Ctrl + 1).
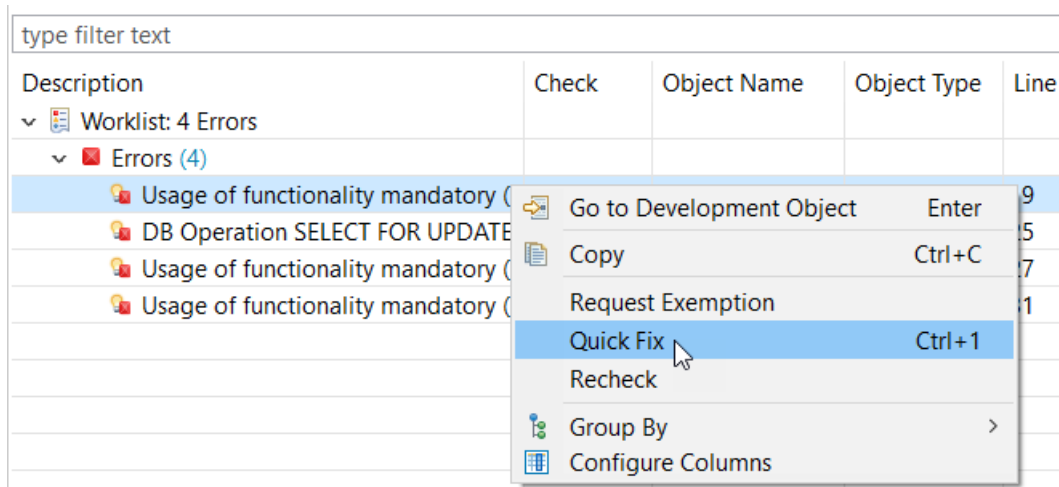
The *Quick Fix* view opens.

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.
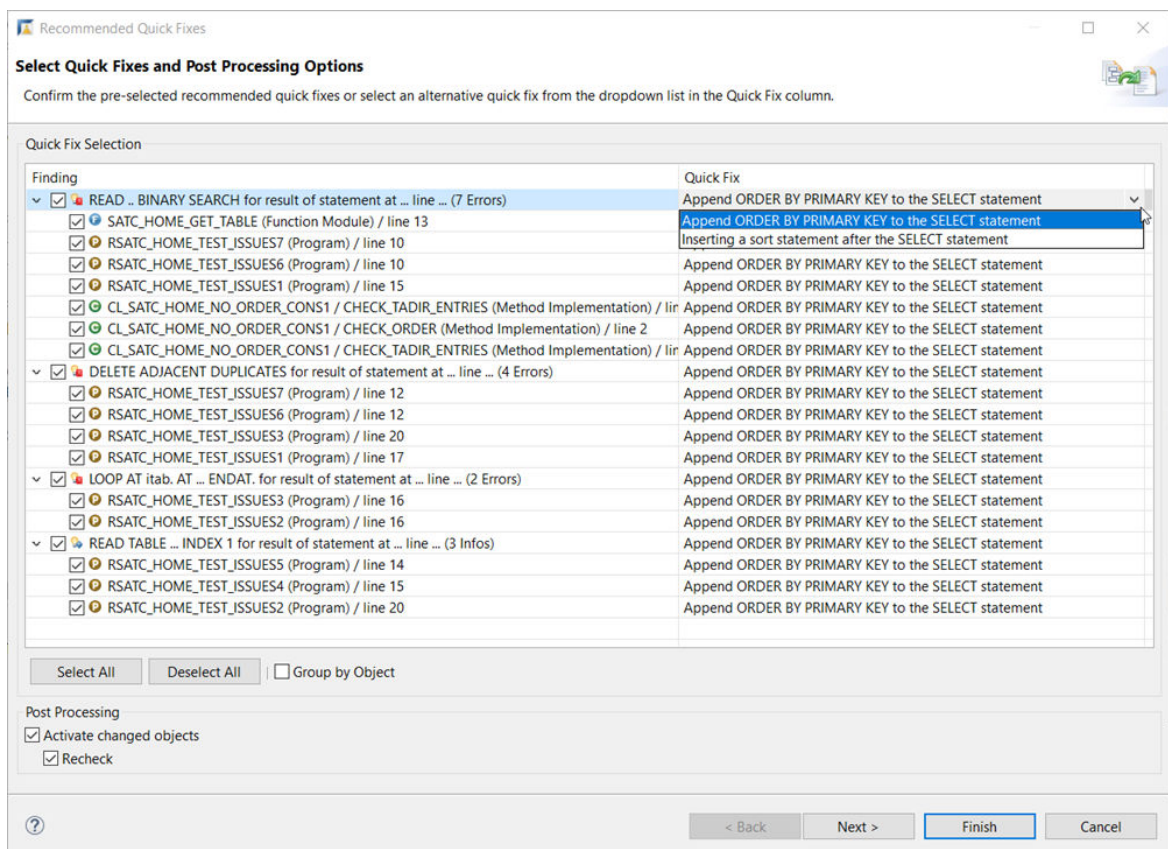
4.3.3.1.1 Applying Recommended Quick Fixes for Multiple ATC Findings

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.

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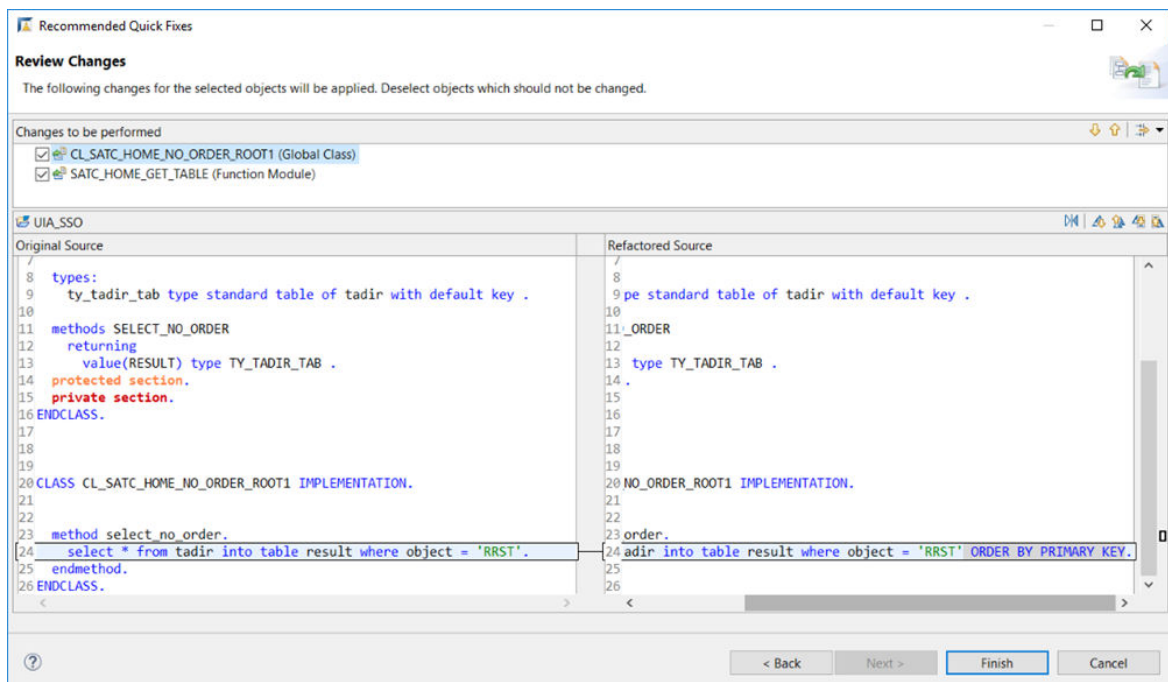
In this example, an `order by primary key` statement was added to the source code to fix the ATC finding.

10. Choose *Finish*.

Results

You applied Quick Fixes for multiple ATC findings at once.

# 4.3.3.2 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

> **i 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 33].
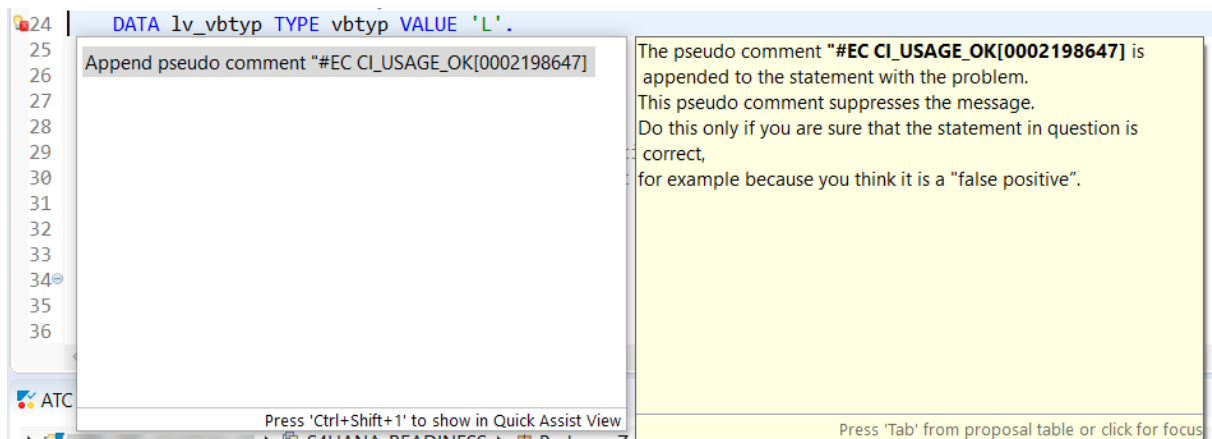
The pseudo comment is appended at the end of the code line in question.

> **→ 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.

4.3.4 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.

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

4.4 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.

5 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.

  > **i 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 the *ABAP SQL Monitor Implementation Guide and Best Practices*: https://www.sap.com/documents/2013/10/92b57ae6-527c-0010-82c7-eda71af511fa.html

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

6   Glossary

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

Term	Definition
ABAP Test Cockpit (ATC)	Tool for performing static and dynamic quality checking of ABAP 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
Central Check System	SAP NetWeaver AS for ABAP 7.52 system (or higher) that checks custom code in one or multiple SAP systems remotely
Checked System	System with custom code that gets checked remotely by the Central Check 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

Important Disclaimers and Legal Information

Hyperlinks

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

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:

    - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
    - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.

- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering a SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

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.

Gender-Related Language

We try not to use gender-specific word forms and formulations. As appropriate for context and readability, SAP may use masculine word forms to refer to all genders.

THE BEST RUN SAP