PUBLIC
SAP Service and Asset Manager
Document Version: 2310 – 2024-03-20

SAP Service and Asset Manager Configuration Guide

© 2024 SAP SE or an SAP affiliate company. All rights reserved.

THE BEST RUN SAP

Content

1 SAP Service and Asset Manager Overview. 8

1.1 Supported Back-End Systems. 8

1.2 SAP Service and Asset Manager Functional Overview. 9

Supported SAP Transaction Codes for SAP Service and Asset Manager. 12

2 SAP Mobile Add-On for the SAP Configuration Panel. 14

2.1 Accessing the SAP Mobile Add-On for SAP Configuration Panel. 14

2.2 Panels, Tabs, and Field Descriptions. 15

Standard Operations in the Configuration Panel. 15

General Info Settings. 16

Geospatial Service Definitions - GIS. 24

Geospatial Service Definitions - GEF. 28

OData Channel Integration Settings. 34

Change Detection Settings. 52

Push Framework Settings. 59

Technical Settings. 67

Mobile Authorization Settings. 68

2.3 Enabling EPD Visualization. 69

2.4 Low Code No Code Support. 72

2.5 Optimized Delta Synchronization. 73

2.6 Shadow Table 74

2.7 Removal of Dependent Objects Based on Lead Objects. 77

2.8 Support for LITE Version of Technical Objects. 79

3 Mobile Add-On Configuration Panel Common Procedures. 83

3.1 Configuring Personas and Features Overview. 83

Supported Features for SAP Service and Asset Manager for Windows. 84

Usage Types. 85

Configuring Personas. 87

Configuring Features. 90

Configuring Component Assignments. 97

Inventory Clerk Parameter Configuration. 98

3.2 Mobile Application Configuration Procedures. 99

Mapping Work Orders to a STARTED Status - Overview. 99

Configuring Auto-Sync. 100

Configuring CATS and PM Confirmation Time Entries. 101

Configuring Clock In Clock Out. 103

Configuring Signature Control. 105
Configuring Namespace Check 106
Digital Signature Overview. 107
Configuring Notification Catalog Types. 112
Enabling and Disabling Features Per User Through SAP Authorization. 114
Configuring Mobile User Attributes. 117
Enabling and Disabling Local Object Mobile Status. 120
Abstract Document Management. 122
EAM Checklist Overview. 135
Configuring the Creation of Technical Objects from a Template. 141
Auto-Resize Images with a Configured Maximum Size. 146
3.3 OData Channel Integration Settings Procedures. 147
Copying an Object to the Customer Namespace. 147
Working with oData MDO Filter Rules. 149
Work Order Distribution by Order Type Overview. 150
Business Object Distribution by Assignment Model. 152
Filtering Equipment Records by Equipment Status. 155
Retrieving Additional Data for OData Mobile Data Objects. 157
Enabling or Disabling Follow-On Work Orders. 159
Maintaining Customer Service Order Types. 160
Maintaining Customer Service Notification Types. 161
Configuring Confirmation Posting Dates - Overview. 162
Configuring the Supervisor Module. 165
Configuring EAM Phase Model - Overview. 175
3.4 Geospatial Service Definitions. 197
Geographic Information System (GIS). 197
Geographical Enablement Framework (GEF). 206
3.5 Push Framework Settings Procedures. 211
Configuring Push for Work Order Assignment Type 1. 211
Configuring Push for Work Order Assignment Type 2. 212
Configuring Push for Notification Assignment Types 1 - 5. 213
Setting up the Outbound Trigger for your Push Configuration. 214
Activating Default Push Services for SAP Service and Asset Manager - Neo. 218
Activating Default Push Services for SAP Service and Asset Manager - Cloud Foundry. 226
Configuring Push for Android. 234

4 Setting up an OData Mobile Data Object. 239
4.1 Supported OData Features for SAP Mobile Add-On. 239
4.2 Setting the OData Mobile Data Object Service Assignment. 241
4.3 Setting the OData Mobile Data Object Configuration. 243
4.4 Setting the OData Model Configuration. 245

5 Configuring SAP Asset Manager for ASPM and PdMS. 251
5.1 Configuring Checklist Parameters. 251
5.2 Setting Up an ASPM Connection to the ASPM System to Use Checklists. 253
Checking the Readiness of the ASPM System 256

6 Field Operations Worker Configuration. 263
6.1 Differentiating a Base SAP Service and Asset Manager Work Order from a Field Operations Worker Order. 263

7 Safety Technician Configuration. 266
7.1 Configuration Prerequisites and Limitations - Safety Technician Persona. 266
7.2 oMDO Configuration. 267
7.3 Mobile Application Parameters. 268
7.4 xChange. 269
7.5 Supported usage types 270
7.6 Work Order as a Dependency 271
7.7 Assignment Management. 272
Assignment Configuration by Assignment Type 4 - Partner Function. 272
Assignment Configuration by Assignment Type 5 - Selection Variant. 278
Assignment Configuration by Assignment Type 6 - Operational List + Selection Variant. 283
7.8 Work Permit Approvals. 287

8 Field Service Technician Configuration on a Customer Service Back End. 291
8.1 SAP Service and Asset Manager Integration with Field Service Management Scheduling. 291
Prerequisites for Configuring Field Service Management. 292
Integration of SAP Service and Asset Manager with Field Service Management Using the Proaxia Field Service Management Connector. 293
Configuring Connectivity between SAP Service and Asset Manager and Field Service Management. 300
Configuring Server Parameters. 301
Configuring the Field Service Management Personas. 304
Status Configuration. 305
8.2 Expense and Mileage Configuration. 307
Configuring Mileage and Expense Parameters. 308
Assignment Type for Planned Expenses and Mileage. 310
8.3 FSM Smartforms. 310
Configuring FSM Smartform DTO Versions 311
Smartform Instance and Template oMDO filters. 312

9 Field Service Technician Configuration on an S/4 Field Service Management Back End 313
9.1 Configuring Connectivity between SSAM and FSM. 313
9.2 Configuring Server Parameters. 315
9.3 Configuring Persona. 317

9.4 Configuring Statuses. . . . 318
9.5 Auto Release of S4 Service Orders. . . . 319

10 Linear Asset Management Configuration. . . . 320
10.1 Disabling and Re-Enabling Linear Asset Management. . . . 320
10.2 Adding Linear Data for Characteristics. . . . 321

11 Meter Management Configuration. . . . 322
11.1 Binding ISU Process Type to Work Order Type. . . . 322
11.2 Binding Meter Reading Notes Based on ISU Process Type. . . . 323
11.3 Meter Reading History Interval Definition. . . . 324
11.4 Binding Meter Reading Reasons Relevant for Technical Installation. . . . 324

12 Quality Management Configuration. . . . 326
12.1 Configuring Quality Management Notification Parameters. . . . 326
12.2 Configuring oData MDO Data Filters for Quality Management. . . . 328

13 SAP Business Application Studio (instead of SAP Web IDE). . . . 331
13.1 Migrating to SAP Business Application Studio . . . 331
13.2 Customizing Apps Using the Mobile Development Kit. . . . 332
13.3 Customizing Metadata Using the Mobile Development Kit. . . . 332
Customizing Contextual Menu Options. . . . 333
Enabling Multi-User Support. . . . 334
13.4 Customizing Deep Linking. . . . 337

14 Data Distribution Model Overview. . . . 342
14.1 Data Distribution by Assignment Types. . . . 343
14.2 Common Filters for SAP Service and Asset Manager. . . . 344
14.3 User Parameter IDs. . . . 346
14.4 Work Order Distribution Rules. . . . 347
Distribution by Work Order Header Person Responsible. . . . 347
Distribution by Work Order Operation Person Responsible. . . . 348
Distribution by Work Order Suboperation Person Responsible. . . . 349
Distribution by Capacity Requirement Person Responsible. . . . 349
Distribution by Work Order Header Planner Group. . . . 350
Distribution by Work Order Operation Work Center. . . . 351
Distribution by Work Order Header Business Partner. . . . 351
Distribution by Work Order Header Work Center. . . . 352
Distribution through MRS Scheduling Engine. . . . 353
Distribution by Free Search. . . . 353
14.5 Notification Distribution Rules. . . . 354
Distribution by Notification Header Person Responsible. . . . 355
Distribution by Notification Task Level Person Responsible. . . . 355

Distribution by Notification Header Level Planner Group. 356
Distribution by Notification Header Level Business Partner. 356
Distribution by Notification Header Level Work Center. 357
Distribution by Free Search - Notifications. 357

15 Troubleshooting. 359
15.1 SAP Gateway Client. 359
15.2 SAP Gateway Error Logs. 360
15.3 SAP Gateway Statistics. 361
15.4 SAP Gateway Tracing Tools. 361

Document History

Before you begin reading this guide, be sure that you have the latest version. Find the latest version at https://help.sap.com/docs/SAP_SERVICE_ASSET_MANAGER.

The following table provides an overview of the most important document changes.

Document Version	Date	Description of Changes
1.0	NOV 2023	Original release of the *SAP Service and Asset Manager Configuration* guide, version 2310

1 SAP Service and Asset Manager Overview

> **Note**
>
> This guide only covers setting up and enabling the SAP Service and Asset Manager application on a Windows platform.

The *SAP Service and Asset Manager Configuration Guide* is intended for system administrators, technical architects, implementation team members, and IT personnel involved in the installation, setup, and configuration of software for the application.

It is assumed that the personnel performing the installation and setup are familiar with SAP installation guidelines. SAP setup knowledge is helpful while carrying out the steps for the setup of SAP.

Use the *SAP Service and Asset Manager Configuration Guide* along with appropriate SAP documentation.

Functions Supported by SAP Service and Asset Manager

SAP Service and Asset Manager is a mobile solution for managing work orders, notifications, condition monitoring, and material consumption. The application also performs time management and failure analysis.

Regardless of connectivity, SAP Service and Asset Manager allows remote employees to access, complete, and manage their assigned work orders and notifications through their devices. With SAP Service and Asset Manager, they have SAP back end data readily available including task lists, repair histories, reference documents, and geospatial data such as addresses and maps. Armed with more information, employees work smarter, have more work time, improve their first-time fix rates, and extend asset lives by conducting more preventative maintenance.

SAP Service and Asset Manager comes packaged with a mobile add-on for SAP ERP and a mobile add-on for SAP S/4HANA. They offer tight integration and easier deployment without interference to or from your existing SAP system customizations or standard SAP objects. They provide you with full configuration, administration, and monitoring features that allow you to manage the SAP Service and Asset Manager application from within your SAP system infrastructure.

1.1 Supported Back-End Systems

SAP Service and Asset Manager supports the following back-end systems:

- SAP S/4HANA Cloud Private edition
- SAP S/4HANA 1610 FPS 03 or higher

> **Note**
>
> For SAP S/4HANA on-premise 1909 systems, no ABAP add-on installation is required. Check 2493602, including the prerequisites section. For the SAP S/4HANA on-premise 1909 release, SAP Service and Asset Manager 1911 is only available in SAP S/4HANA 1909 FPS01 and above releases.

- SAP Enhancement Package 7 for SAP ERP 6.0 Support Package 14 or higher
- SAP Enhancement Package 8 for SAP ERP 6.0 Support Package SP07 or higher

> **Note**
>
> With the SAP S/4HANA Cloud Public edition back-end system, you must use SAP Maintenace Assistant.

The same functions are available for both back-end systems.

For detailed information, see the following guides:

- Mobile Add-On for ERP Installation Guide
- Mobile Add-On for S/4HANA Installation Guide

1.2 SAP Service and Asset Manager Functional Overview

SAP Service and Asset Manager is a mobile solution for managing work orders, notifications, condition monitoring, and material consumption. The application also performs time management and failure analysis.

Regardless of connectivity, SAP Service and Asset Manager allows remote employees to access, complete, and manage their assigned work orders and notifications through their mobile devices. With SAP Service and Asset Manager, they have SAP back end data readily available including task lists, repair histories, reference documents, and geospatial data such as addresses and maps. Armed with more information, employees work smarter, have more work time, improve their first-time fix rates, and extend asset lives by conducting more preventative maintenance.

SAP Service and Asset Manager comes packaged with a mobile add-on for SAP ERP and a mobile add-on for SAP S/4HANA. They offer tight integration and easier deployment without interference to or from your existing SAP system customizations or standard SAP objects. They provide you with full configuration, administration, and monitoring features that allow you to manage the SAP Service and Asset Manager application from within your SAP system infrastructure.

The main features and functions available in SAP Service and Asset Manager include the following.

Work Orders and Notifications

SAP Service and Asset Manager supports the following standard SAP Plant Maintenance Work order functionalities on the mobile device:

- Download Work order assignments based on various assignment models
- Display work orders and their details, including repair histories

- Edit work orders and their details
- Create work orders and their details
- Complete work orders and update their status

The following standard SAP notification functionalities are supported on the mobile device:

- Download notification assignments based on various assignment models
- Display notifications and their details
- Edit notifications and their details
- Create notifications and their details
- Complete notifications and update their status

Maintenance Execution Data Capture

The following data related to maintenance execution can be captured from the mobile device:

- Maintenance order confirmation, goods issue for work orders
- Measurement readings for work orders
- Damage codes for notifications

Time Management

Maintenance technicians can use SAP Service and Asset Manager to trace their time efficiently and accurately by entering the timesheet and the attendance records from the mobile device.

Single Sign-On (SSO)

Single Sign-On (SSO) allows the user to log into the SAP Service and Asset Manager application from the client using SSO credentials without having to enter their back end user name and password. In addition, once logged in with SSO, you can access another mobile application without the need to log in again.

Documents

SAP Service and Asset Manager supports viewing of leading data or transaction data attachments on the mobile device. Documents include Microsoft Office files, PDFs, and other commonly used business documents, including videos, pictures, and audio files.

Downloading and uploading documents are supported for the following objects:

- Work orders
- Notifications

- Equipment
- Functional locations

The following standard SAP document storage options are supported:

- SAP Business Document Service (BDS)
- SAP Generic Object Service (GOS)
- SAP Document Management System (DMS)

Field Operations Worker Component

Inspection rounds with routing is an optional feature available as the SAP Service and Asset Manager Field Operations Worker (FOW) component. FOW supports:

- Route and stop definitions
- Technical object assignments
- Measuring point assignments
- Measurement readings

Route and stop definition is implemented via the standard work order inspection round functionality.

Crew Management Component

SAP Service and Asset Manager supports work crew management. This feature supports:

- Daily crew list definition
- Crew member and vehicle assignment
- Crew time management

Crew management is an optional feature available as the SAP Service and Asset Manager Crew Management component.

Meter Management Component

SAP Service and Asset Manager supports the industry solution for utilities meter management. The following standard features are supported:

- Meter installation (full or technical) via work order
- Meter replacement via work order
- Meter removal via work order
- Meter repair via work order

Customer Service Component

SAP Service and Asset Manager supports customer service. This feature supports:

- Details of service engagements of the technician with the customer
- Details of business partners for the customer
- Technicians have access to contract and warranty information for the customer
- Mapping functionality for both customer addresses and partner address, provided the mobile device has internet access

Asset Central Component

Asset Central links production systems and assets with manufacturing and maintenance business processes to reduce operational and maintenance costs and increase asset uptime. Using Asset Central, you can use PdMS, or Predictive Maintenance and Service equipment indicators that allow you to identify the health status of your equipment.

1.2.1 Supported SAP Transaction Codes for SAP Service and Asset Manager

SAP Service and Asset Manager uses the SAP back end and specific SAP ERP transaction codes to help configure the application.

Transaction Codes	Definition
IW21	Create notifications
IW22	Edit notifications
IW23	Display notifications
IW31	Create work orders
IW32	Edit work orders
IW33	Display work orders
IW51	Create service notification
IW52	Change service notification
IW53	Display service notification
CAT2	Record time entries

Transaction Codes	Definition
IE03	Display equipment BOM
IE13	Display functional location BOM
IL03	Display functional location
MM03	Display material
IK13	Display measurement document
IK12	Change measurement document
IK11	Create measurement document
IW41	Create PM confirmation
MIGO	Goods issue for work orders
CV01N	Create DMS document
CV03N	Display DMS document
OAOR	Manage BDS document

2 SAP Mobile Add-On for the SAP Configuration Panel

The SAP Mobile Add-On provides integration services for SAP Service and Asset Manager. A central configuration tool known as the SAP Configuration Panel is provided to perform all configuration tasks related for the mobile application. The Configuration Panel is a browser-based application based on Web Dynpro ABAP.

2.1 Accessing the SAP Mobile Add-On for SAP Configuration Panel

Context

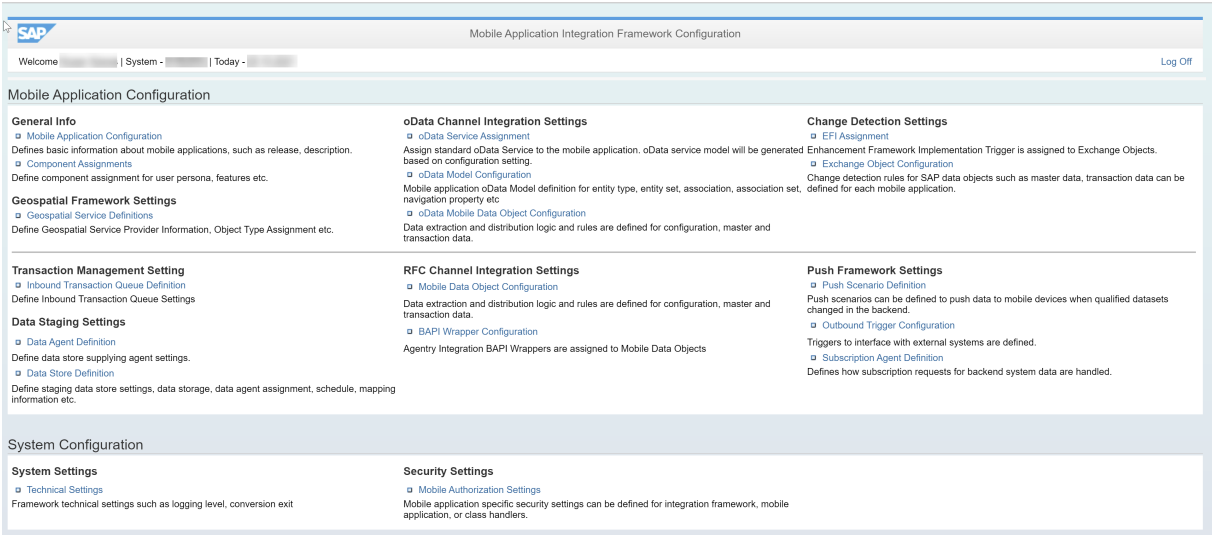
You can access the Configuration Panel either through SAP Customizing or using a transaction code directly. First, log into your back-end system, and then you can choose from the following two options:

Procedure

1. To access the ConfigPanel through *Customizing*, enter the transaction *spro* to open the *Customizing: Execute Project* screen. Select the *SAP Reference IMG* tab. Using the *SAP Customizing Implementation Guide* list, select *Agentry SAP Framework Configuration* > *System Settings* > *Define Mobile Applications*.
2. To access the ConfigPanel using a direct transaction code shortcut, enter */n/syclo/configpanel*.

Results

The *Mobile Application Integration Framework* home page is displayed in a browser.



2.2 Panels, Tabs, and Field Descriptions

Perform configuration activities for your app through the ConfigPanel.

Customization changes you make via the ConfigPanel can significantly impact the behavior of your app. Always follow SAP best practices, make changes and test them in the development and quality control systems before you transport the changes into your production landscape.

2.2.1 Standard Operations in the Configuration Panel

While configuration for each mobile application is unique, certain toolbar functions in the Configuration Panel are common and are available for all applications.

Mobile Application Filter

If more than one mobile application is available in the same system, you can use the filter function to only view items for a specific application. Find the filter option on any page where multiple applications are displayed.

To filter by application, click the arrow to the right of the *Defined Mobile Applications* field, and select the appropriate mobile application. To remove the selection and view all items for all mobile applications in the system, click in the field again and select the asterisk (*) symbol.

Creating, Copying, Deleting, and Changing Items

The following standard actions are available to configure different components and items within your mobile application setup:

- **Create:** Creates a new item. All modifiable fields are empty.
- **Copy:** Copies the item that was highlighted and creates a new item. All modifiable fields are filled in with the information from the existing item and are available for changes before saving.
- **Delete:** Deletes the highlighted item.
- **Change:** Allows you to change the highlighted item in the modifiable fields.

Saving or Canceling Changes for an Item

Once you click the *Create*, *Copy*, or *Change* button, the *Save* and *Cancel* buttons are displayed. After you change the configuration of the item, click *Save* to save the changes or *Cancel* to discard the changes.

> **Note**
>
> If the *Save* and *Cancel* buttons are active, the *Home* link for the ConfigPanel is not available. Either save your changes or cancel out of the changes to return to the main Configuration Panel page

Message List

Certain actions can generate system messages. These messages can be error messages or informational messages. If you perform an action that prompts a system message, a message bar appears above the main panel with a brief description of the message.

Click the *Show List* button to display the detailed view of the message list.

2.2.2 General Info Settings

The following areas are used in configuring general information for the application:

-
-

2.2.2.1 Mobile Application Configuration

The Mobile Application Configuration page allows you to configure general settings for the entire mobile application.

The *Mobile Application Configuration* page contains the following tabs:

- General
- Mobile Status Setting
- Conversion Exit Setting (not used in SAP Service and Asset Manager)
- System Components (not used in SAP Service and Asset Manager)
- Parameters
- Client Globals (not used in SAP Service and Asset Manager)
- User Attributes (not used in SAP Service and Asset Manager)
  Application Persona
- Features

General Tab

Use the *General* tab to create or change basic information about a mobile application.

- **Basic Data section:** Enter the name of the mobile application in the `<Mobile Application>` field, which is limited to 40 characters. Select the type of application in the `<Type>` field. Note that for SAP Service and Asset Manager, the type is `<oData Applications>`. Enter a brief, easy to understand description in the `<Description>` field, limited to 60 characters. Type in the release number of the application in the `<Release>` field.
- **User Management Setting:** When the `<Disable Automatic User Creation>` box is checked, a new user GUID is not automatically created when a new mobile client is detected in the system. Manually create and maintain mobile users through the Administration portal.
- **Server Management Setting:** When the `<Disable Automatic Server Registration>` box is checked, a new server GUID is not automatically created when a new server is detected in the system. You must manually create and maintain servers through the Administration portal.
- **Life-cycle management:** When the `<Application Blocked>` box is checked, the mobile application is disabled. The mobile user can no longer connect to the back-end system for the mobile application, and the xChange process is also disabled. The `<Effective Date>` and `<Time>` fields flag when the change takes effect.
- **xChange Setting:** When the `<Disable Change Detection>` box is checked, the change detection process, or xchange process, for the application is completely disabled.
- **Inbound Transaction Management:** Not used in the SAP Service and Asset Manager application.
- **Multi Backend Setting:** When checked, enables a specific mobile application to connect to multiple SAP systems, consisting of one host server and one or more satellite servers.
- **System Role:** Dropdown menu where you can select either *Host* or *Satellite*.
  A *Host* system is the connection between SAP and the SAP Service and Asset Manager application in the SAP Business Technology Platform. The host server provides the logic to the application and functions as the bridge to the satellite server or servers. There can only be one host server per system.

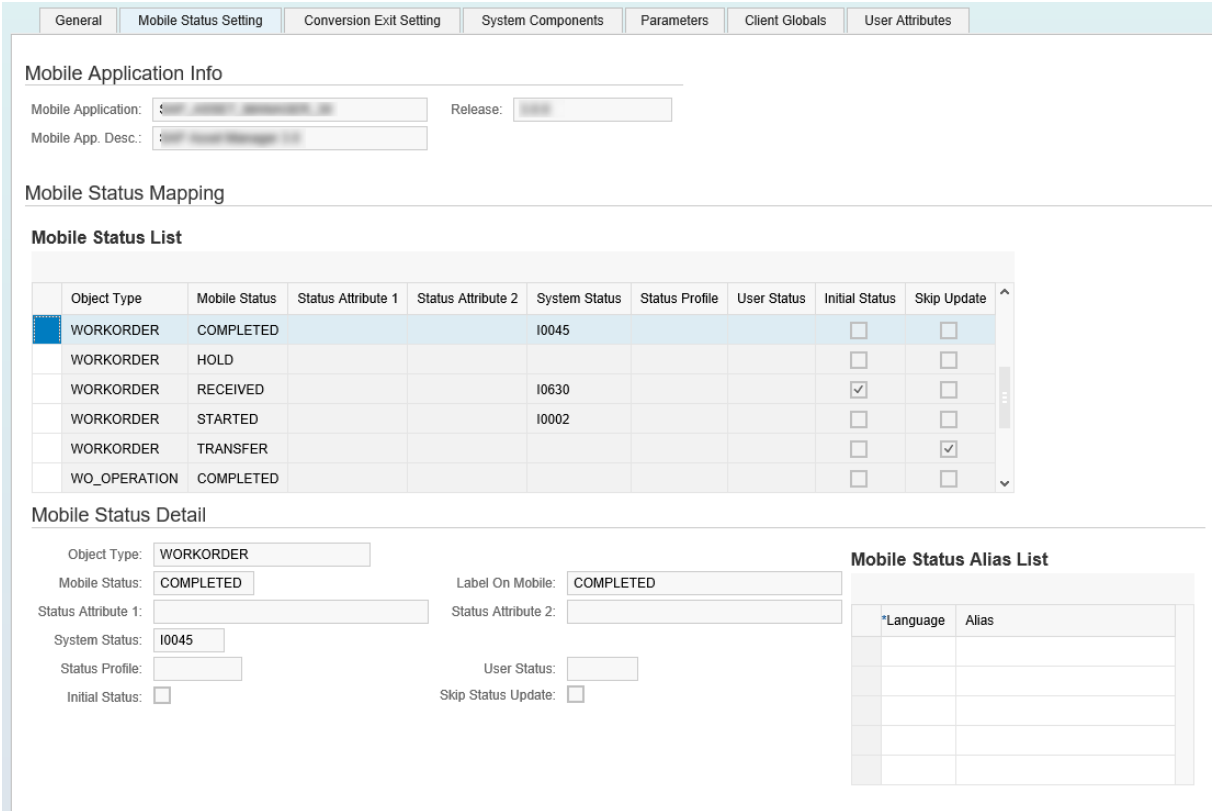
Satellite servers communicate with SAP through the host server. To complete multi backend configuration, configure the host and back-end servers using the *System Components* tab. See that section for more details.

Mobile Status Setting Tab

Use the Mobile Status Setting tab to map the available mobile statuses that an oData mobile data object (OMDO) supports on the client side. If a user status also exists for the same object type, you can link it to the mobile status and the system status through this tab.

- **Mobile Application Info:** The `<Mobile Application>` field is read only and is the name of the mobile application. The `<Mobile Application Description>` is read only and is a brief description of the mobile application. The `<Release>` field is read only and is the release number of the application.
- **Mobile Status Mapping:** Use the `<Add Status>` and `<Delete Status>` buttons to create and delete mobile status mappings. Fill out the `<Object Type>` with the specific object in the mobile application, for example, `<Notification>`. The `<Mobile Status>` is the status defined by the mobile application. The `<Label on Mobile>` is not used. The `<User Status>` is an SAP status code as defined in SAP. Note that the status codes are language independent codes.
  If the `<Initial Status>` checkbox is selected, the mobile status is displayed by default when you download the object to the mobile device. To skip a specific mobile status update from a mobile device, use the `<Skip Update>` checkbox corresponding to the mobile status object.
  Use the *Mobile Status Alias List* table to define language-specific mobile status aliases.

In the following example screen, the highlighted row in the mapping table indicates that if a user sets a work order to completed, the application sets the work order system status to I0045 in SAP.



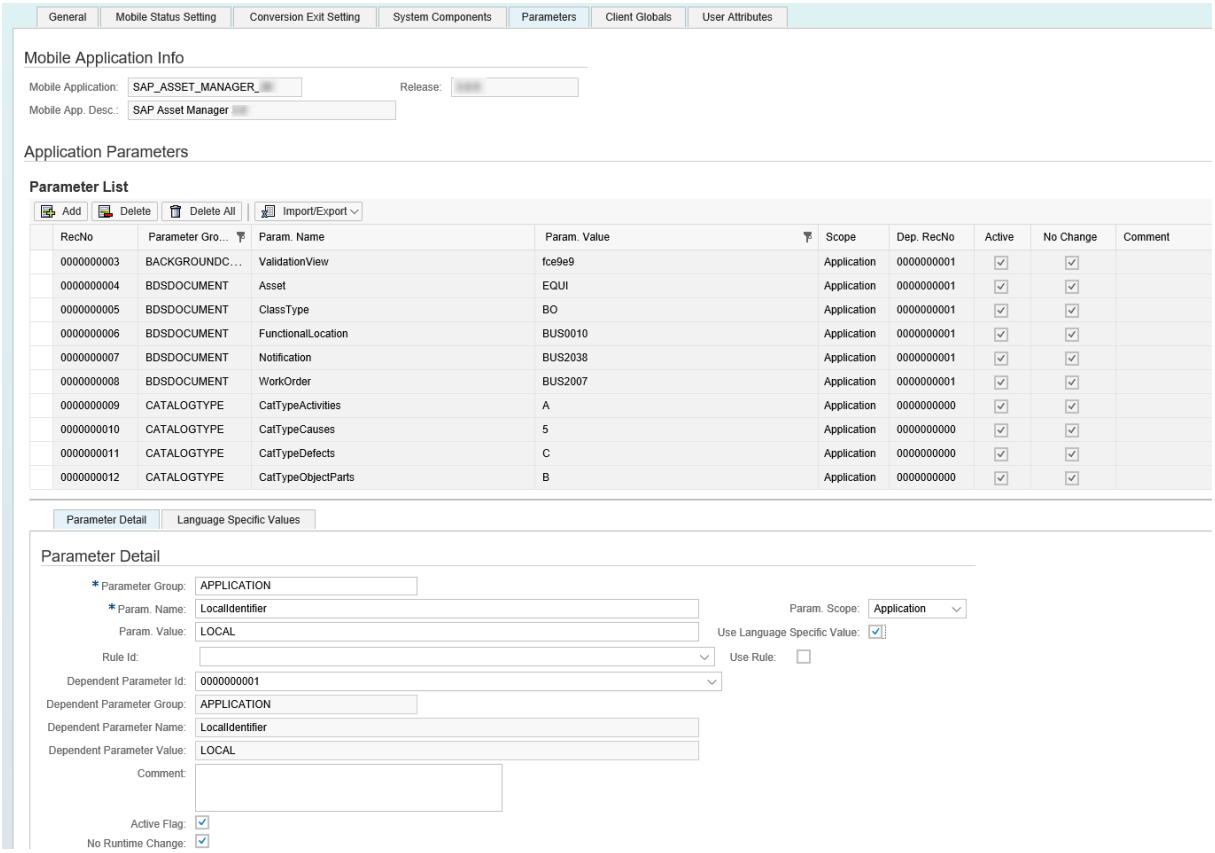
If there is no system status or user status, the mobile status only affects the mobile device and does not affect the backend SAP system.

If there is a user status specified but no status profile when the mobile user sets the mobile status, the app sets that user status for the object, disregarding the status profile of that object.

If there is a user status and status profile specified when the mobile user sets the mobile status, the app sets that user status if the object uses that status profile.

Parameters Tab

The *Parameters* tab defines system parameters.



- **Mobile Application Info:** The `<Mobile Application>` field is read only and is the name of the mobile application. The `<Mobile Application Description>` is read only and is a brief description of the mobile application. The `<Release>` field is read only and is the release number of the application.
- **Application Parameters:** Use the `<Add>` and `<Delete>` buttons to create and delete parameters.
- **Parameter Detail:** The `<Parameter Group>` is the group to which the parameter belongs. Groups are how you organize parameters. References to a parameter include both the group name and the parameter name. The `<Parameter Name>` is the unique name of the parameter.
  The `<Parameter Value>` is the currently configured value of the parameter. References to the parameter return the configured value. Use the `<Language Specific Value>` checkbox to select which parameters you wish to be language dependent. The checkbox and the corresponding *Language Specific Values* tab are only active after you have clicked the *Change* button. Note that the language available in the

Language Specific Values tab, *Value List* table, depends on the language you are using to log into the mobile client. You must manually maintain each parameter that you wish to control for a language specific value separately.
Set the `<Parameter Scope>` to one of the following options:

- Mobile Application: Value for all users of the application
- Mobile User: Value you can override for individual users. To override a parameter value of a user, see the Administration & Monitoring Portal on parameters

> **Note**
>
> For information on setting user parameters, see the following security guides, depending on your back end system:
>
> - Mobile Add-On for S/4HANA Security Guide
> - Mobile Add-On for ERP Security Guide

The `<Rule ID>` field contains the rule used at runtime. If you check the `<Use Rule>` box, the rule in the `<Rule ID>` field is active.
Check the `<Active Flag>` box to ensure that the parameter is used by the mobile application. Inactive parameters are not used by the application. When you check the `<No Runtime Change>` box, you cannot override the value of the parameter. The configured value is always the value. If the box is not checked, the parameter values can be overridden at runtime through synchronization processing.

Application Persona Tab

The persona selected determines the data that is downloaded to the mobile client.

Mobile Application Info

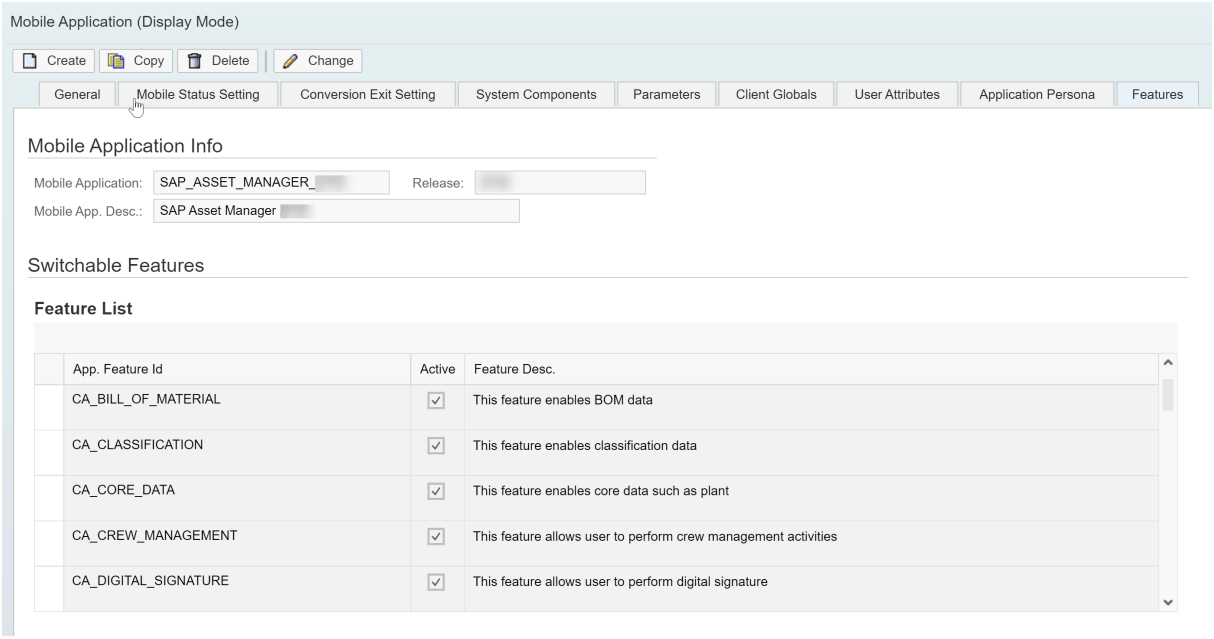
Read-only information about the application.

Application Personas

- Auto Determination Rule:
- Persona List: The following personas are standard in an SAP Service and Asset Manager installation:
  - MAINTENANCE_TECHNICIAN (default)
  - INVENTORY_CLERK

Features Tab

Switchable features allow you to configure various components into features. Feature assignment determines the data that is downloaded to the mobile client.



Mobile Application Info

Read-only information about the application.

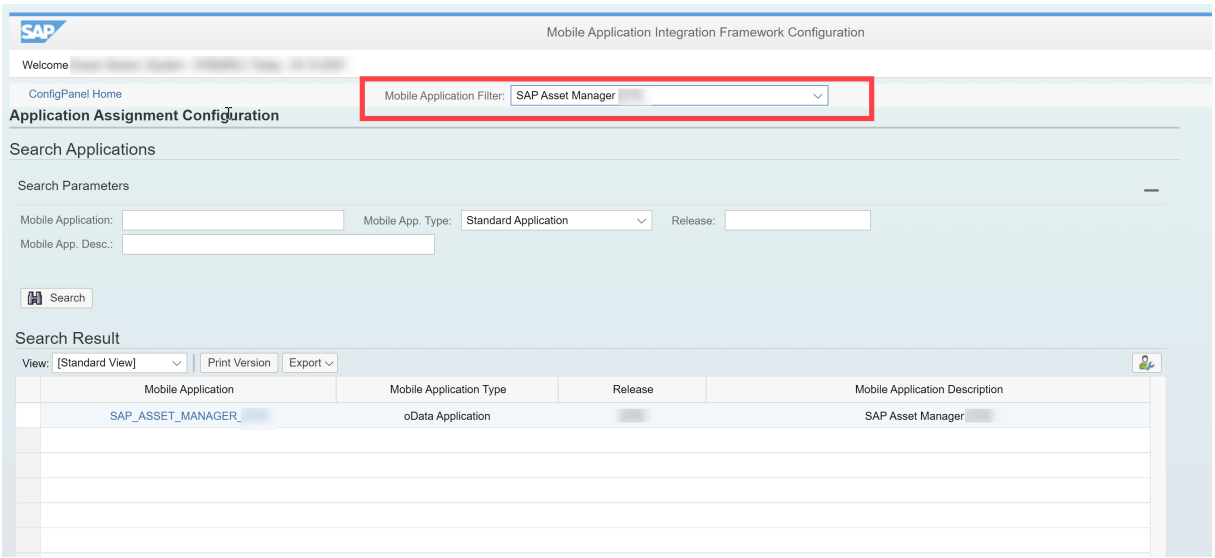
Switchable Features

Check or uncheck a feature from the feature list to enable or disable it.

2.2.2.2 Component Assignments

The *Component Assignments* page allows you to configure persona and feature assignments.

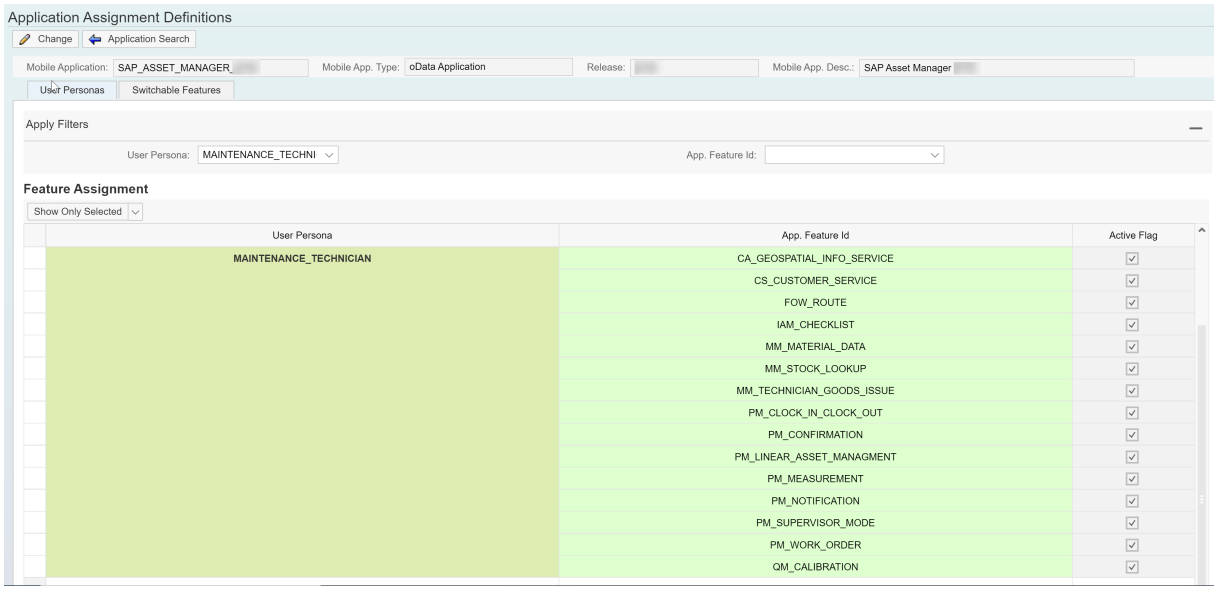
Use the *Mobile Application Filter* field to select your application. Then, click the application hyperlink in the *Search Results* table.



User Personas Tab

Use the *Apply Filters* section to filter for a specific persona or specific features. In the following example, we've filtered for all features that belong to the *MAINTENANCE_TECHNICIAN* persona.

The *Feature Assignment* section shows you the filtered results. Use the drop-down menu to further filter your selections. For example, we've chosen to display only active, or selected features for the maintenance technician persona.



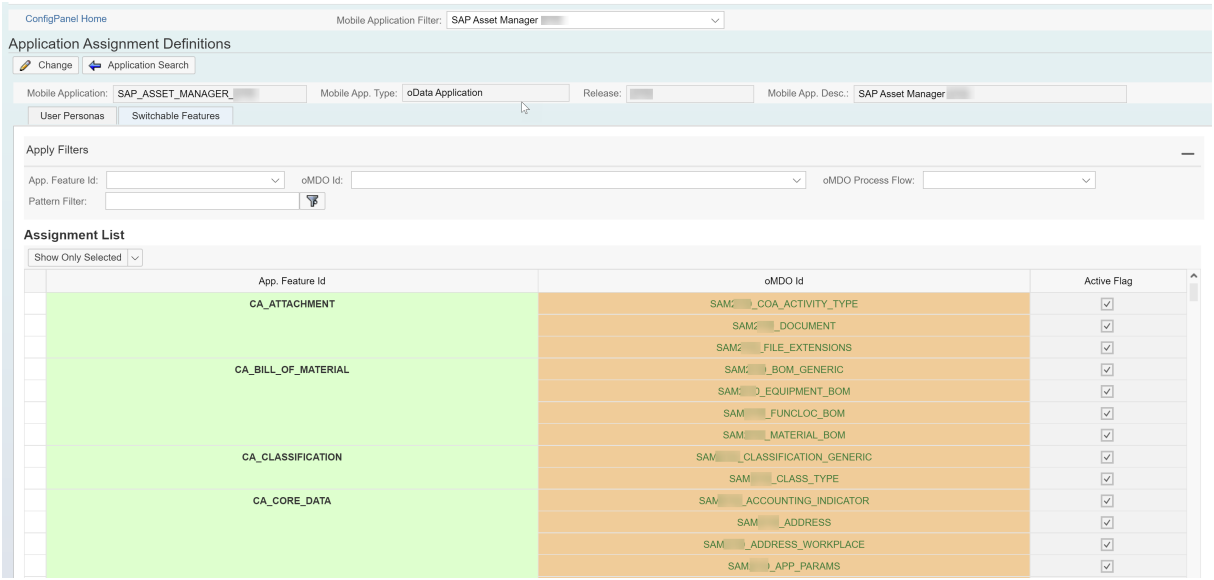
Switchable Features Tab

The *Switchable Features* tab allows you to configure OMDOs required for each feature. During initial and delta syncs, the mobile client only downloads data from the assigned OMDOs.

Use the *Apply Filters* section to filter for a specific OMDO or feature ID.

The *Assignment List* section shows you which OMDOs are assigned and active (or not active) with a feature.

> **Note**
>
> Don't select OMDOs that belong to the online service (ex: */MERP/SAP_ONLINE_LOOKUP_EXT_<version>*). These online service entities might not exist in the base service (*/MERP/SAP_ASSET_MANAGER_<version>*).



2.2.2.3 Mobile Transaction History

With the *Mobile Transaction History* feature, SAP introduces the possibility for users to record all successful *CREATE* and *UPDATE* transactions from the mobile device in to the */SYCLO/MBL02* table. This provides insight into which objects are either created or changed via the mobile application, by whom, and when the changes are made.

To enable mobile transaction history logging for your mobile app, use one of the following options:

1. On the Mobile Application Configuration page (transaction `/n/SYCLO/CONFIGPANEL`), select the *Enable Mobile Transaction History* check box.
2. Run the program `/SYCLO/CORE_RUNTIME_PARAM_PROG`, select your mobile application and the parameter *ENABLE_MTRAN_HIST*.

> **Note**
>
> If you specify a user ID, only transactions for that user will be logged. Otherwise, the runtime parameter will be applied to all users using the application.

To view and search mobile transaction history in the Syclo admin panel (transaction `/n/SYCLO/ADMIN`) through the *Mobile Transaction History Monitor* application:

- Open the transaction, click the *Monitoring* tab and select *Mobile Transaction History Monitor*
- Select a mobile application and view the mobile transaction history for this application.

> **Note**
>
> Use the *oMDO Id* and *Technical Entity Type* filters to filter by object type, and the *Object Key* filter to display individual objects.

> ### Note
>
> To remove records from the *Mobile Transaction History* table, you can use the `/SYCLO/CORE_MTRAHIS_PURGE_PROG` report.

2.2.3 Geospatial Service Definitions - GIS

A geographic information system (GIS) integrates hardware, software, and data for capturing, managing, analyzing, and displaying all forms of geographically referenced information.

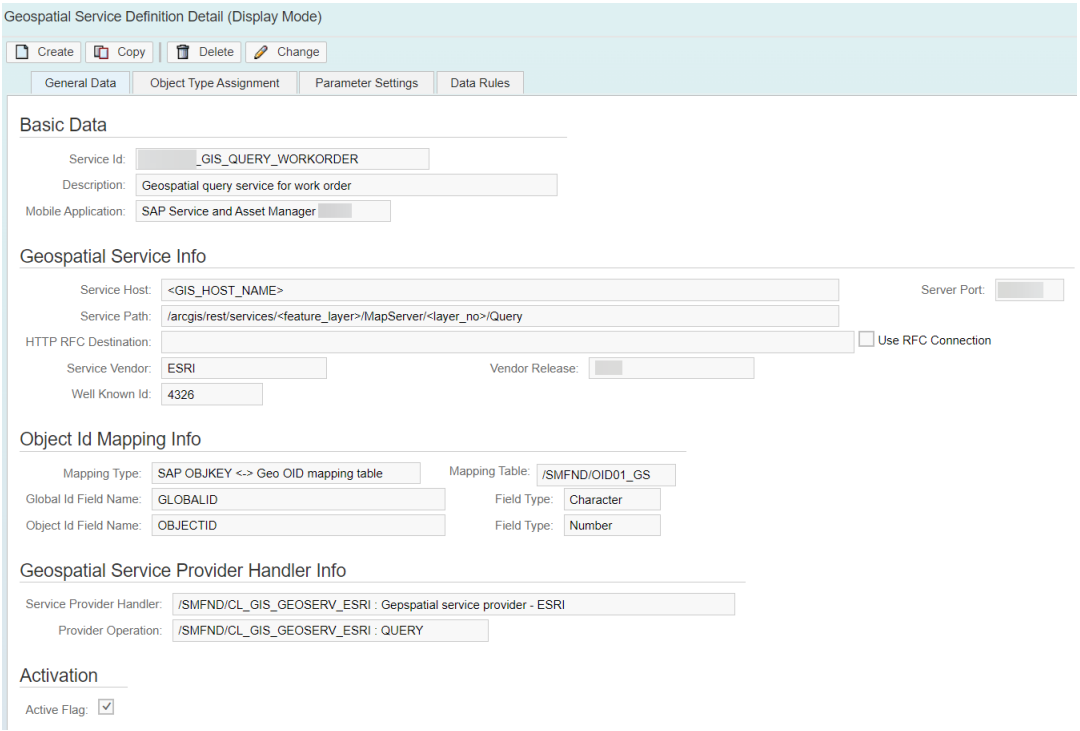
> ### Note
>
> GIS is enabled by default. See the topic if you are using GEF (SAP Geographical Enablement Framework).

Geospatial data plays an important part in the daily operations of many organizations. By adding geospatial data to the technical data of an asset, you get a full picture of that asset.

The *Geospatial Service Definitions* page contains the following tabs:

- General Data
- Object Type Assignment
- Parameter Settings
- Data Rules

General Data Tab



General Data tab:

Basic Data

- **Service ID:** Required field. Name of the geospatial service ID, limited to 40 characters, with namespace protection. Use the Y or Z namespace.
- **Description:** Description of the geospatial service
- **Mobile Application:** Mobile application of the geospatial service. Every geospatial service is assigned to a specific mobile application.

Geospatial Service Info

- **Service Host:** Host name of the geospatial service provider

  > **Note**
  >
  > The information in this field is used in the `Target Host` field in the Creating and Configuring an RFC Destination for Offline Maps [page 203] procedure.

- **Server Port:** Port number of the geospatial service provider
- **Service Path:** End-point URL of the geospatial service
- **HTTP RFC Destination:** RFC destination pointing to the GIS. By using the RFC destination, you can support various logon authentication methods of the geospatial service. If you use an RFC destination, service host, port, and path are not required.
- **Use RFC Destination:** Indicator that you must use an RFC destination to connect to GIS
- **Service Vendor:** Vendor name of the GIS provider.SAP uses Esri for its out of the box installation of the SAP Service and Asset Manager application.
- **Vendor Release** Version information of the GIS provider. The Esri version for SAP Service and Asset Manager 2310 is *ArcGIS iOS SDK 100.2.1* for iOS and *ArcGIS Runtime SDK for Android 100.3.0* for Android.
- **Well Known ID:** Well-known ID of the coordinate system

Object ID Mapping Info

- **Mapping Type:** Defines how the SAP object ID is mapped to the object ID in the geospatial service provider database. Select from the following:
  - **No mapping:** SAP object ID is the same as the object ID in the GIS provider database
  - **SAP OBJKEY <-> GEO OID-mapping table:** Mapping is stored in a mapping table
  - **Lookup Routing:** Use a lookup ABAP object-oriented class to determine the mapping dynamically. The ABAP object-oriented lookup class must implement the `/SMFND/IF_GIS_OID_MAPPING` interface.
- **Mapping Table:** SAP mapping table name, if used. The standard `/SMFND/OID01_GS` mapping table is the default. The standard `/SMFND/GIS_OIDMAP_UPLOAD_PROG` program is provided to load the table with a text file
- **Object ID Lookup Routine:** OID lookup routine name, if used. Used in conjunction with the *Lookup Routing* mapping type
- **Global ID Field Name:** Name of the Global ID field in the GIS provider database, if relevant. For example, in the Esri database, the global field name is *GLOBALID*
- **Object ID Field Name:** Name of the `<Object ID>` field in the GIS provider database, if relevant. For example, in the Esri database, the field name is *OBJECTID*
- **Field Type** Use the dropdown to select the type of field for both the `<Global ID>` and the `<Object ID>`
- **Service Provider Handler:** ABAP object-oriented class that handles integration to the GIS provider

- **Provider Operation:** ABAP OO class operation that handles integration to the GIS service provider.

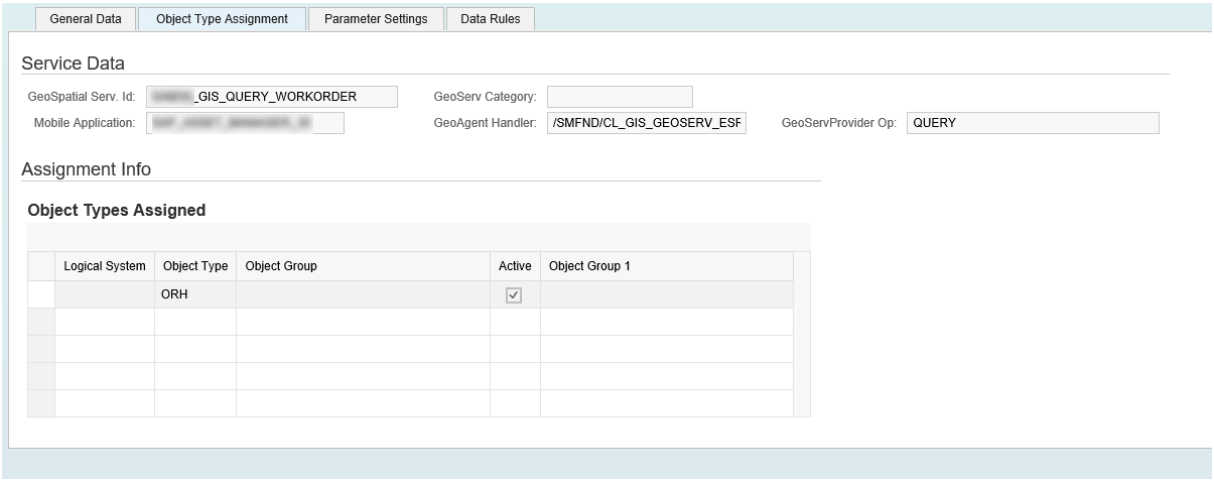
Activation

- **Active Flag** Check the checkbox to activate the GIS query service

Object Type Assignment Tab

Use the *Object Type Assignment* tab to define what type of SAP objects are assigned to the geospatial service. You can define different geospatial services for different SAP object types.

For example, you can map equipment with polygon geospatial data to one geospatial feature layer. You can then map equipment with point geospatial data to a different geospatial feature layer.

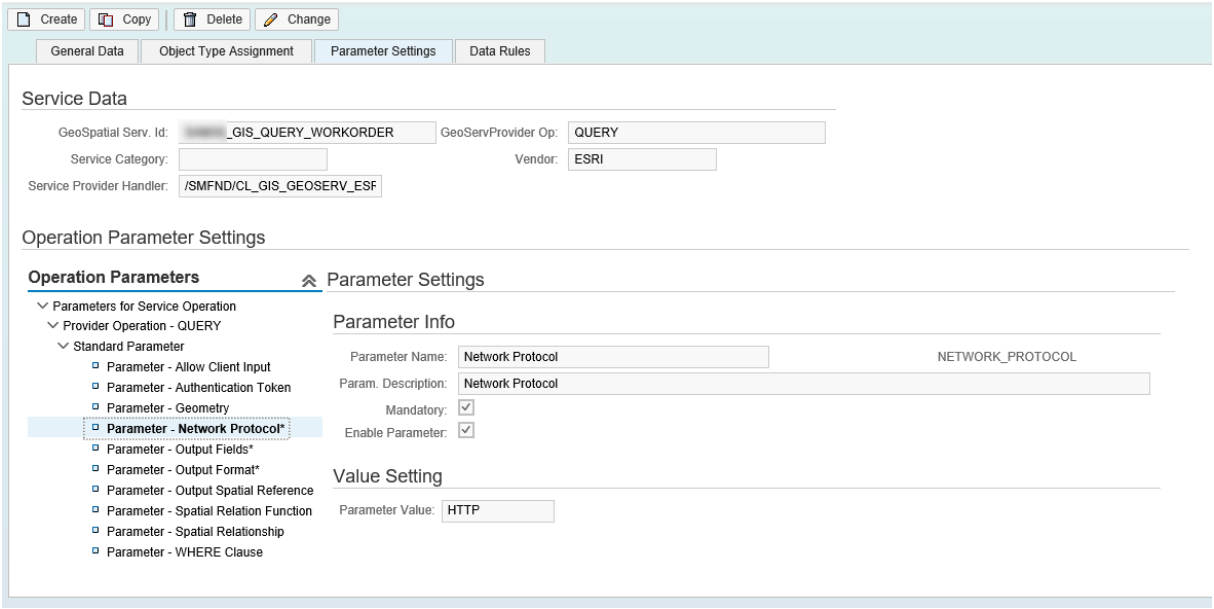


Assignment Info

- **Logical System:** Logical system of the SAP object. A logical system is required to properly identify the SAP object if the mobile add-on aggregates data from different back-end systems.
- **Object Type:** Type of the object as it is identified in the mobile add-on. For example, the standard object type *IEQ* is used to identify the *Equipment* object.
- **Object Group:** Optional setting used to further group the objects in the same object type.
- **Active:** When the *Active* checkbox is marked, the assignment is active.
- **Object Group 1:** Optional setting used to further group objects of the same object type and object group.

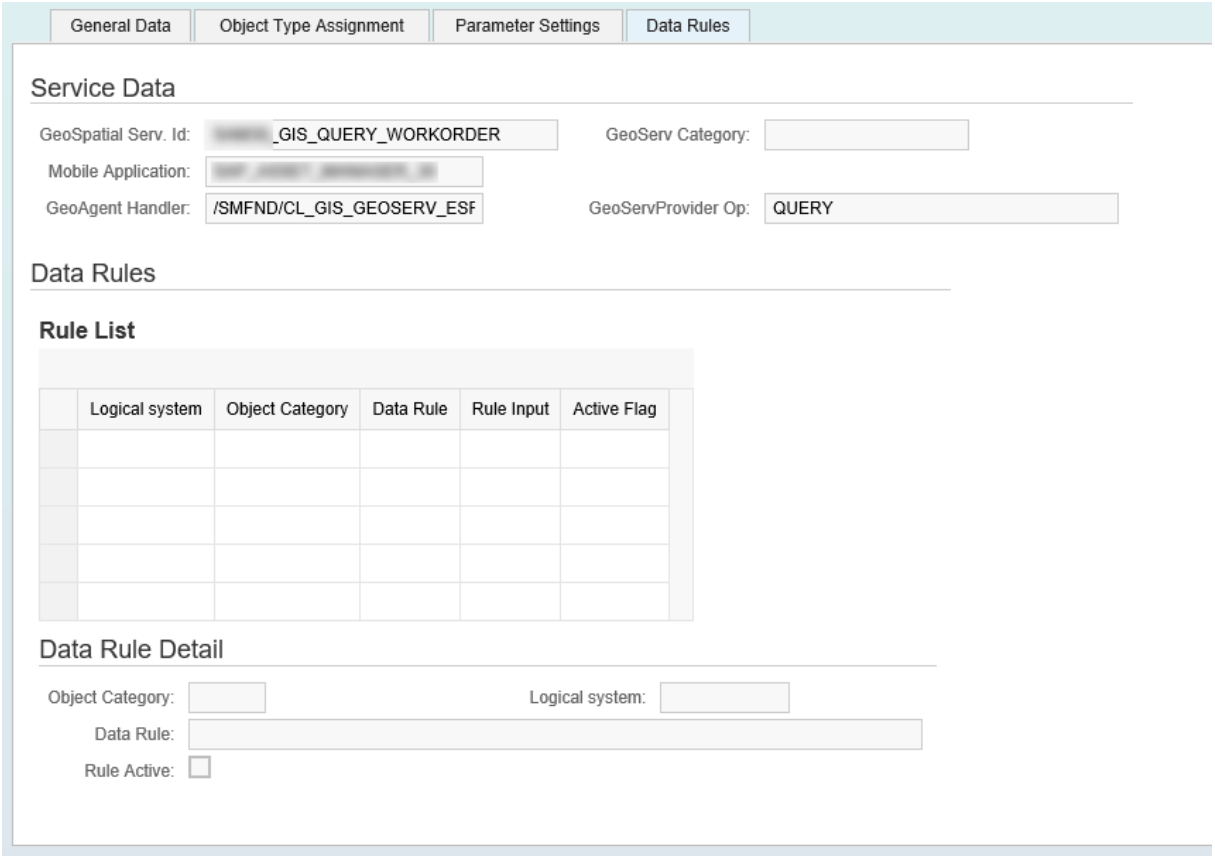
Parameter Settings Tab

Use the *Parameter Settings* tab to define parameter settings for the service provider handler. The service provider handler can declare the list of parameters that might require input. If parameters are declared, they are displayed on this tab, and you can enter values for them.



Data Rules Tab

Use the *Data Rules* tab to define data rules. A data rule is used to transform input data to the service provider handler, before calling the geospatial service. For example, to dynamically assign values of object type, object group, and object group 1 to input data, use a data rule. Using a data rule influences which geospatial service is assigned to an input object.



Data Rule Detail

- **Object Category:** Type of the object as it is identified in the mobile add-on. For example, you can use the standard object type *IEQ* to identify the *Equipment* object.
- **Logical System:** Logical system of the SAP object. If the mobile add-on aggregates data from different back-end systems, the logical system is required to identify the object.
- **Data Rule:** Data rules are ABAP OO classes that implement the `/SMFND/IF_GIS_DATA_RULE` interface.
- **Rule Active:** If the checkbox is checked, the rule is active.

2.2.4 Geospatial Service Definitions - GEF

The SAP Geographical Enablement Framework (GEF) enables the augmentation of business data with spatial attributes for SAP S/4HANA applications. The framework allows SAP data to be used in GIS-based geo-processing operations.

> **Note**
>
> GIS is enabled by default. See the topic if you're using GIS (Geographical Information System).

GEF reduces, and in some cases, eliminates the need for complex synchronization between SAP and GIS systems. Business data can be combined with engineering data in a single map view independent of the user working with SAP tools or GIS tools, decreasing TCO, increasing the value of business data, and simplifying user interaction.

The framework provides an embeddable map-based UI for SAP S/4HANA applications to quickly geo-enable their business objects and support geospatial processes. It also exposes the geometries and attributes of geo-enabled SAP business objects as feature classes to be consumed via standard GIS tools.

The *Geospatial Service Definitions* page contains the following tabs:

- General Data
- Object Type Assignment
- Parameter Settings
- Data Rules

General Data Tab

As GEF is integrated directly to SAP S/4HANA systems, there's no need to call REST APIs outside of SAP S/4HANA. All information and coordinates are already stored as part of the system.

General Data tab:

Basic Data

- **Service ID:** Required field. Name of the geospatial service ID, limited to 40 characters, with namespace protection. Use the Y or Z namespace.
- **Description:** Description of the geospatial service
- **Mobile Application:** Mobile application of the geospatial service. Every geospatial service is assigned to a specific mobile application.

Geospatial Service Info

- **Service Host:** Automatically set to *NO_HOST_NAME*
- **Server Port:** Port number of the geospatial service provider
- **Service Path:** Automatically set to *NO_SERVICE_PATH*
- **HTTP RFC Destination:** RFC destination pointing to the GIS. By using the RFC destination, you can support various logon authentication methods of the geospatial service. If you use an RFC destination, service host, port, and path are not required.
- **Use RFC Destination:** Indicator that you must use an RFC destination to connect to GIS
- **Service Vendor:** Vendor name of the GIS provider. For GEF, automatically set to SAP.
- **Vendor Release** Not required for GEF
- **Well Known ID:** Not required for GEF

Object ID Mapping Info

- **Mapping Type:** Automatically set to *No Mapping*. Objects are directly integrated to GEF.
- **Global ID Field Name:** Not used for GEF
- **Object ID Field Name:** Automatically set to *GEF_OBJECTID*. Only used for visibility into which field is used.
- **Field Type** Use the dropdown to select the type of field for both the `<Global ID>` and the `<Object ID>`
- **Service Provider Handler:** Automatically populated with `/MERP/CL_CORE_GIS_GEOSERV_GEF`

> **Note**
>
> Custom handlers are supported.

- **Provider Operation:** Automatically populated with `/MERP/CL_CORE_GIS_GEOSERV_GEF`

Activation

- **Active Flag** Check the checkbox to activate the GEF query service

Object Type Assignment Tab

Use the *Object Type Assignment* tab to define what type of SAP objects are assigned to the geospatial service. You can define different geospatial services for different SAP object types.

For example, you can map equipment with polygon geospatial data to one geospatial feature layer. You can then map equipment with point geospatial data to a different geospatial feature layer.

Assignment Info

- **Logical System:** Logical system of the SAP object. A logical system is required to properly identify the SAP object if the mobile add-on aggregates data from different back-end systems. You can configure the following fields on the
- **Object Type:** Type of the object as it is identified in the mobile add-on. For example, the standard object type *IEQ* is used to identify the *Equipment* object.
- **Object Group:** Optional setting used to further group the objects in the same object type.
- **Active:** When the *Active* checkbox is marked, the assignment is active.
- **Object Group 1:** Optional setting used to further group objects of the same object type and object group.

Parameter Settings Tab

Use the *Parameter Settings* tab to define parameter settings for the service provider handler. The service provider handler can declare the list of parameters that might require input. If parameters are declared, they're displayed on this tab, and you can enter values for them.

There are two standard parameters: EAM Scenario and Output Format

The *EAM Scenario* parameter data mines the GEF scenario object. For each scenario, there's a corresponding business object:

EAM Scenario ID	GEF Business Object ID
EAM Equipment EAMS_EQUI	EAMEQUI
EAM Functional Location EAMS_FLOC	EAMFLOC
EAM Maintenance Notification EAMS_NOTIF	EAMNTF
EAM Maintenance Order EAMS_ORDER	EAMORD

You can assign different business objects for your system needs.

Output Format: The only supported parameter value is JSON.

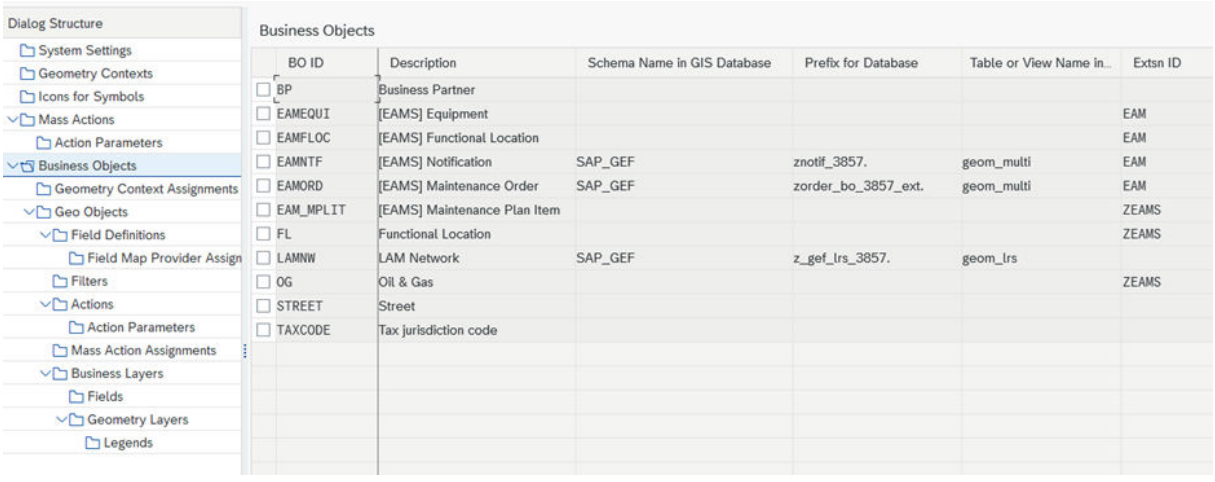
Data Rules Tab

The *Data Rules* tab isn't used in GEF configuration.

2.2.4.1 Integrating Mobile Application Integration Framework (MAIF) with SAP Geographical Enablement Framework (GEF)

The integration point between the *Mobile Application Integration Framework (MAIF)* and *SAP Geographical Enablement Framework (GEF)* has been improved to include the ability to call geometry on the corresponding `geo_id` of a business object. Previously, this capability was not available and MAIF would call all geometries on each business object. This is undesirable because there may be a large amount of data stored at the business object level for users, and only some of it is relevant to day-to-day operations. This enhancement assumes that our users have GEF data segregated at the geo_id level.

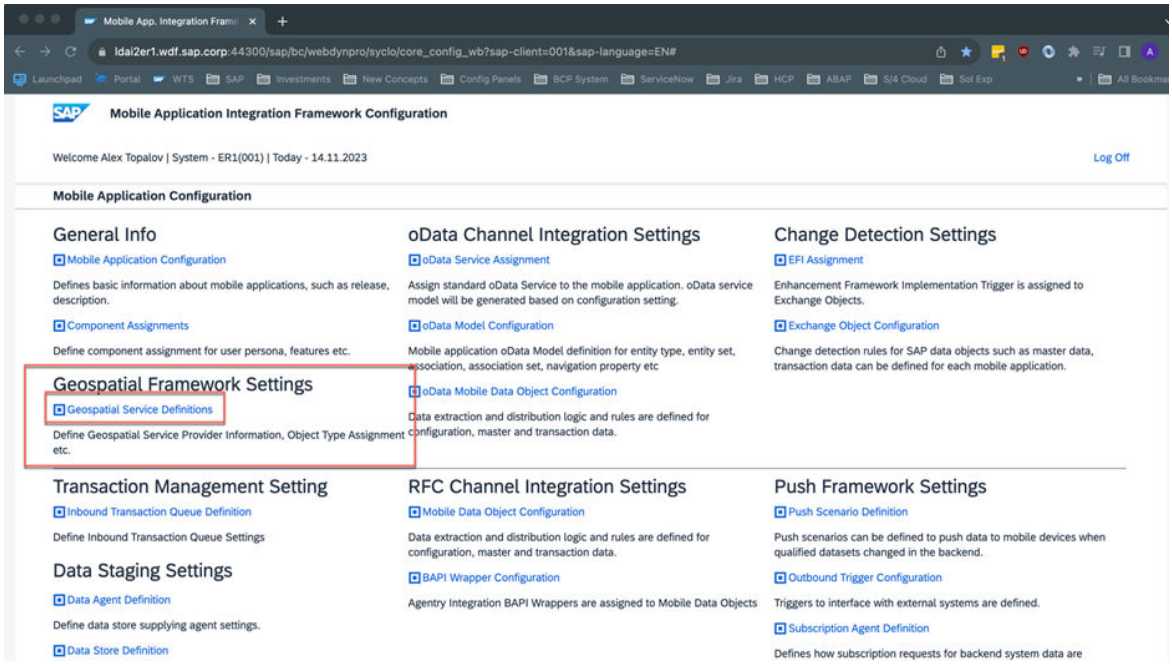
The backend configuration is maintained in the `GEF_FRW_CONFIG GEF` (framework) configuration transaction. Selection of business objects from the hierarchy. This opens the maintained SAP business objects, such as work orders, equipment.



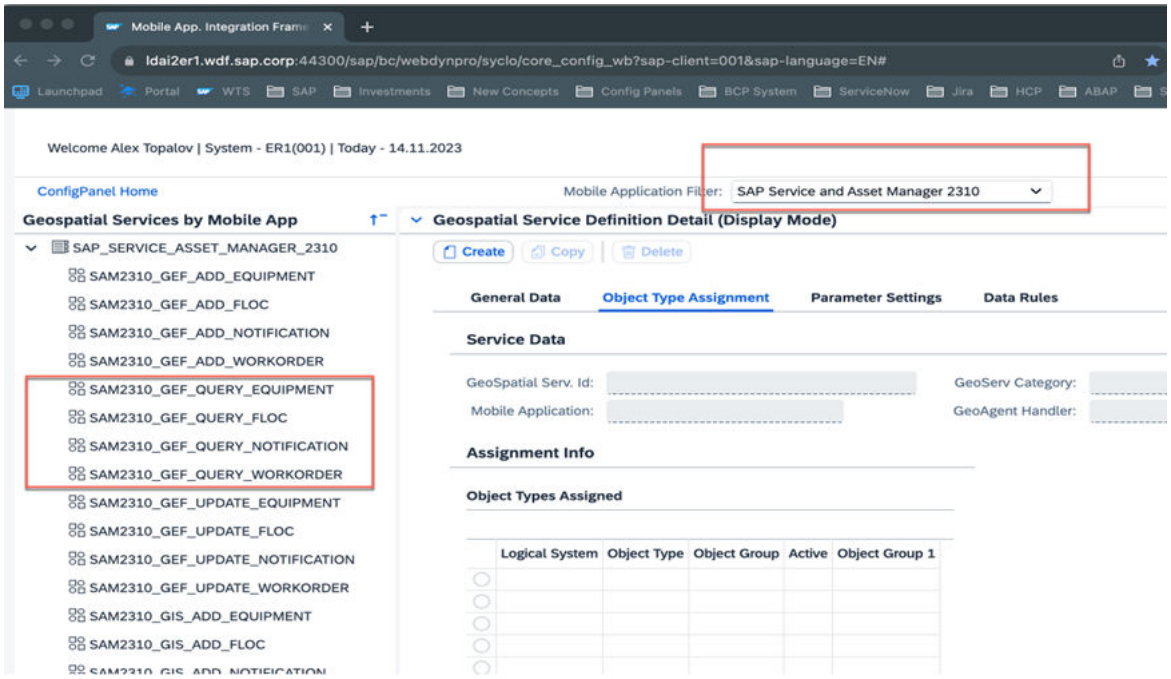
1. Select a business object, for the example `EAMEQUI Equipment`.
2. Once the row is selected, click on `Geo Objects` (sub-level) in the hierarchy.
3. The `Geo Objects` configuration is where the configuration for each equipment type (for the given example) is listed. Each of the geo-objects is a separate sub-type that can maintain geodata. Some of these geo-objects are not relevant to the mobile use case and must not be downloaded to the mobile device.

> ### Note
>
> There are additional settings in the field and filter definitions in geo-objects (sub-hierarchy) that will exclude object instances from being fetched from the databases if these fields and filters are not met (this is not a MAIF specific configuration and must be maintained properly).
>
> > ### Note
> >
> > Additional configuration in MAIF is only maintained in the *Geospatial Service Definitions* section under the SAMXX_GEF_QUERY_XXX (XX (at the beginning) represents the mobile application version and XX (at the end) represents the business object at play). The current example is for `SAM2310_GEF_QUERY_EQUIPMENT`.

4. Navigate to the configuration panel and click on Geospatial Service Definition.

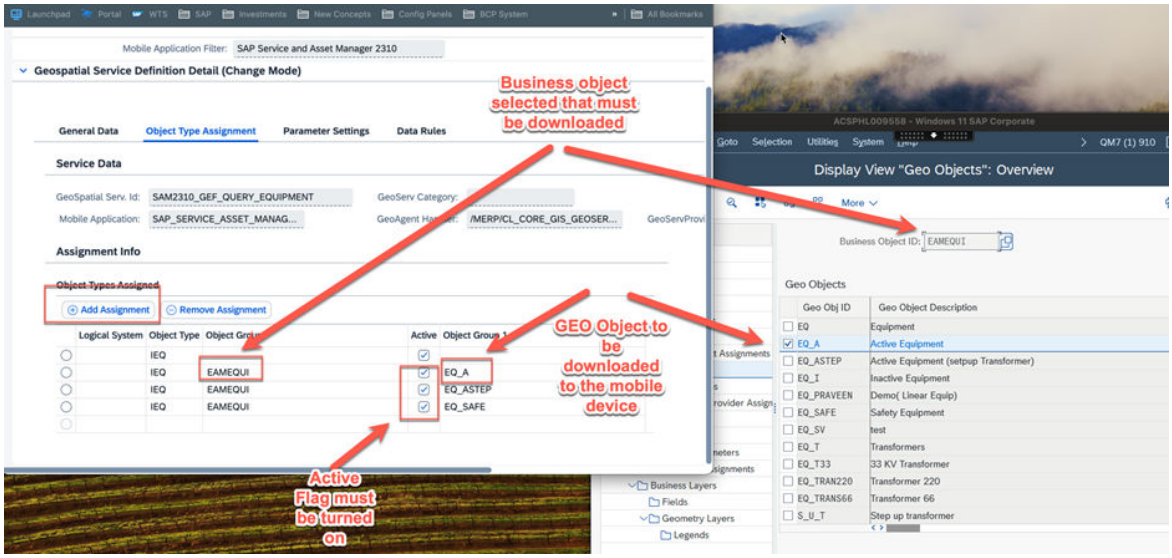


5. Select the mobile application filter for the mobile application you will modify, in this case SAP Asset Manager 2310, and open the full list of objects.
   The configuration will only be applicable for `GEF_QUERY` geospatial services.

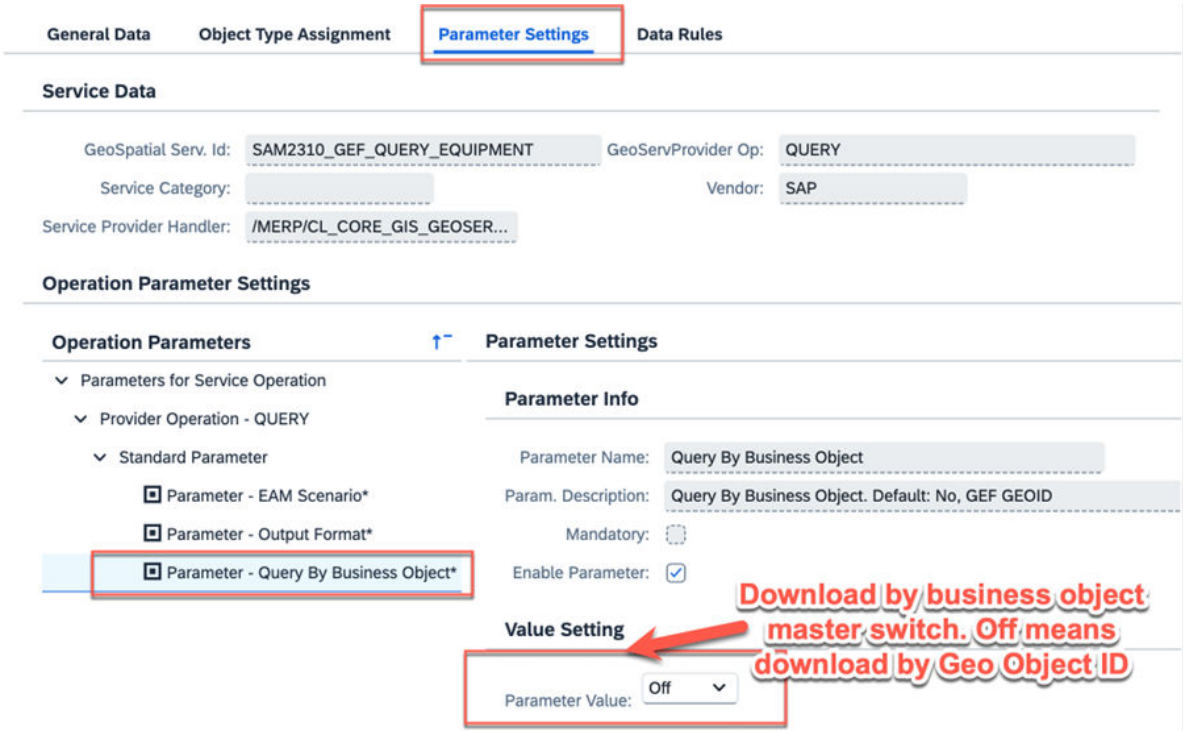


6. Click on `SAM2310_GEF_QUERY_EQUIPMENT` and select the *Object Type Assignment* tab. This table will allow replication of the geo_id which will be attempted to be downloaded.

> **Note**
>
> To successfully complete the download, all relevant geo_ids must be listed along with the corresponding business objects, and this still depends on the filters and fields in the GEF Framework configuration, which is the source of truth for the backend configuration.



7. Once the desired configuration has been replicated, go to the *Parameter Settings* tab and make sure that the global switch for downloading by business object is turned off.
   As a result, MAIF will stop downloading all geometries based on business objects only, and will strive to download objects by Geo Object ID, according to the replicated Geo Objects listed in Object Type Assignment as configured in this guide above



This configuration of geo-object identifiers must be replicated for work orders, functional locations, notifications, and equipment to ensure that geospatial data is downloaded for each of these objects.

2.2.5 OData Channel Integration Settings

2.2.5.1 OData Service Assignment

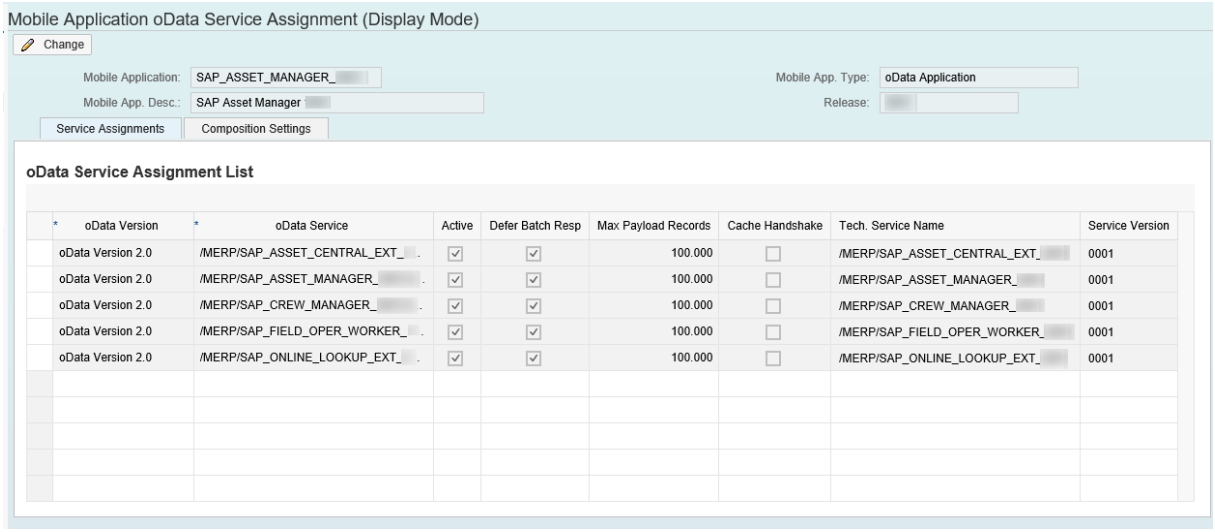
Gateway OData services implemented using the Mobile Integration Framework for SAP are different from the typical Gateway OData services.

The following requirements must be met for the Gateway OData services:

- Define the Gateway OData technical model using the generic model provider class of the Mobile Integration Framework `/MFND/CL_CORE_ODATA_V2_MPC`. You can maintain the OData technical model with transaction `/IWBEP/REG_MODEL`.
- Define the Gateway OData technical service using the generic data provider class of the Mobile Integration Framework `/MFND/CL_CORE_ODATA_V2_DPC`. You can maintain the OData technical service with transaction `/IWBEP/REG_SERVICE`.
- Assign the Gateway OData technical service to a mobile application by choosing the *OData Service Assignment* in the ConfigPanel.
- Do not define the Gateway OData technical model using the Gateway Service Builder. The model is determined and generated dynamically by the generic model provider class `/MFND/CL_CORE_ODATA_V2_MPC` based on the model configuration settings defined in the ConfigPanel.

- The generic data provider class `/MFND/CL_CORE_ODATA_V2_DPC` doesn't provide the required business logic for the Gateway OData technical service. Business logic is provided by OMDOs. Assign every OData business request to the service to an OMDO. The assigned OMDO performs the necessary business logic for the business request.

Service Assignments



You can define the following settings for the OData service assignment:

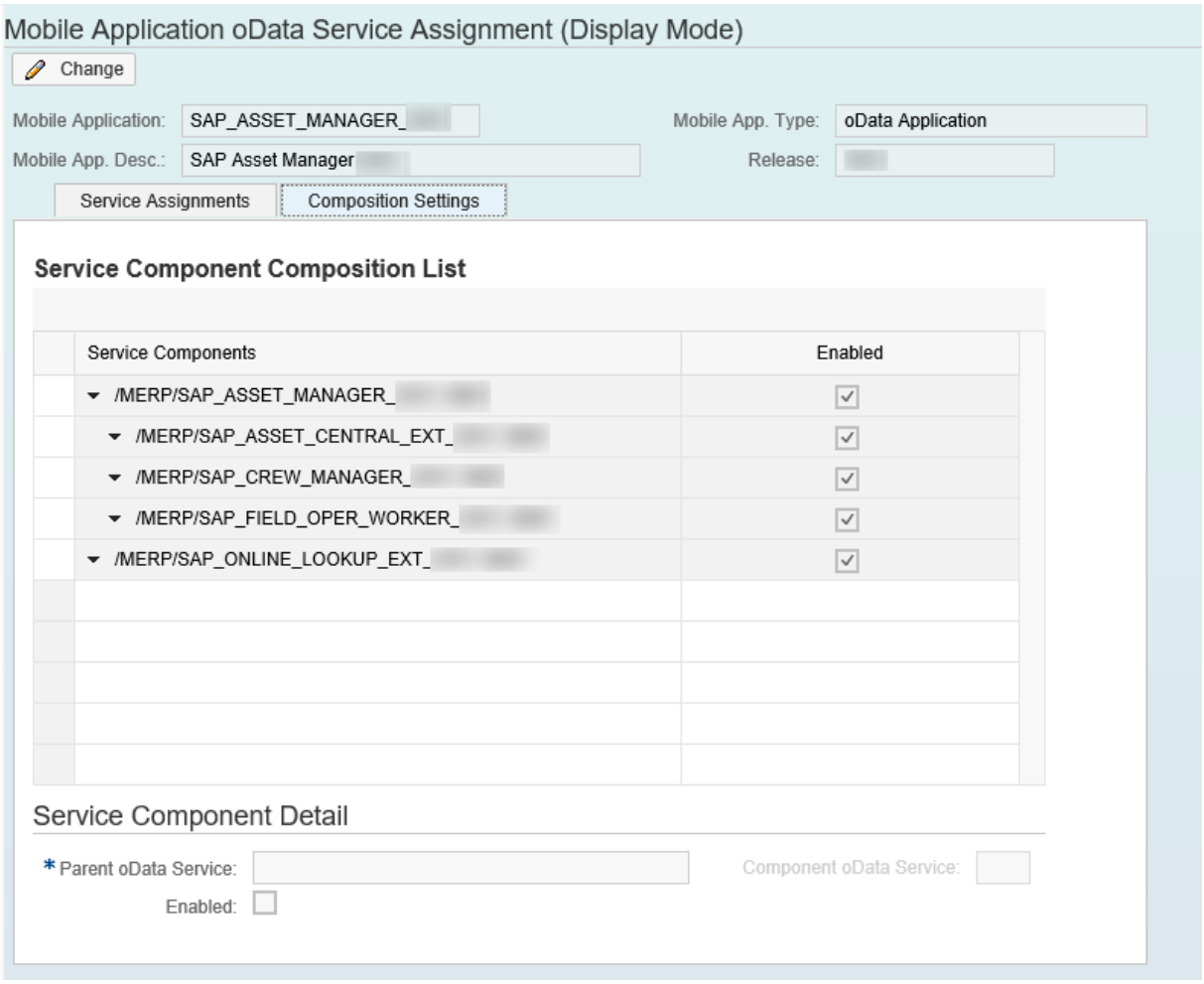
- **OData Version:** OData version 2.0 is supported.
- **OData Service:** The Gateway OData technical service that is assigned to the mobile application. You can only assign a single mobile application to a Gateway OData technical service. Assignment to multiple mobile applications is not supported.
- **Active:** If the *Active* checkbox is checked, both the OData model and OData service are supported. If the checkbox is not checked, no OData model is generated for the service, and the data request to the service is not supported.
- **Defer Batch Response:** This setting is only relevant if you have an SAP S/4HANA back-end system. If the checkbox is checked, the Gateway runtime deferred batch response is enabled, which can improve performance during read request processing.
- **Max Payload Records:** Maximum number of records in the response payload to a read request. In case the read request is a batch request with multiple read requests, this setting defines the total number of records allowed in the overall payload, which is the aggregation of individual read request responses.
- **Cache Handshake:** When checked, the Gateway runtime metadata cache handshake is enabled for the service.
- **Technical Service Name:** Read-only information. Gateway OData technical service name.
- **Service Version:** Read-only information. Gateway OData technical service version.

Composition Settings

With service component composition, you can compose a complex service using component services.

In the following example, service `/MERP/SAP_ASSET_MANAGER_1911 version 0001` is composed of service `/MERP/SAP_ASSET_CENTRAL_EXT_1911 version 0001`, service `/MERP/SAP_CREW_MANAGER_1911 version 0001`, and service `/MERP/SAP_FIELD_OPER_WORKER_1911 version 0001`.

The entity model for service `/MERP/SAP_ASSET_MANAGER_1911 version 0001` includes the entity model from `/MERP/SAP_CREW_MANAGER_1911 version 0001` and the other two entity models listed. The entity model for service `/MERP/SAP_CREW_MANAGER_1911 version 0001` only contains its own entities.



To define a component composition, define the following:

- **Parent OData Service and Version:** Parent OData service. Entity model of a child OData service is included in the parent entity model. Association and navigation properties can be defined between parent service and component service.
- **Component OData Service and Version:** Child OData service
- **Enabled:** If the checkbox is not checked, the entity model of the component service is not included in the entity model of the parent service.

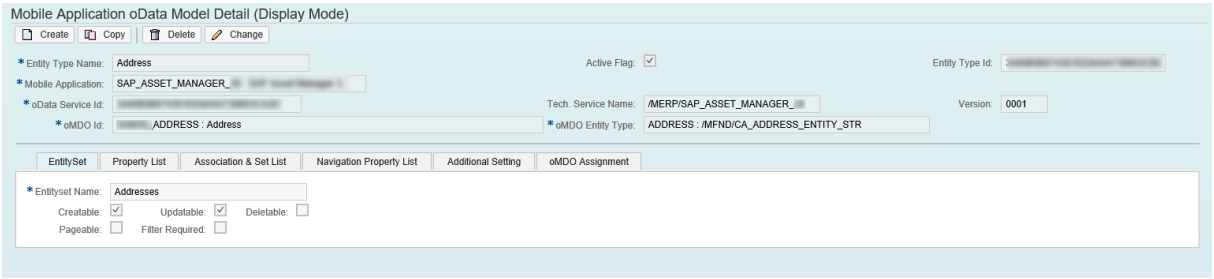
2.2.5.2 OData Model Configuration

OData service implemented using the Mobile Application Integration Framework does not use the Gateway Service Builder to define the OData model. Define the OData model using the OData Model configuration tool in the ConfigPanel. The runtime OData model is generated dynamically based on the configuration settings. The OData model configuration is mobile application-specific. You cannot share OData models across mobile applications.

Define the OData model configuration settings through the following screens:

Entity Set Tab

Entity configuration defines the OData entity type. Entity set configuration defines the OData entity set. In an OData model configuration, each entity type is limited to one entity set. Reuse of entity types by multiple entity sets or by different OData services is not supported.



The following attributes are available for the *Entity Type* definition:

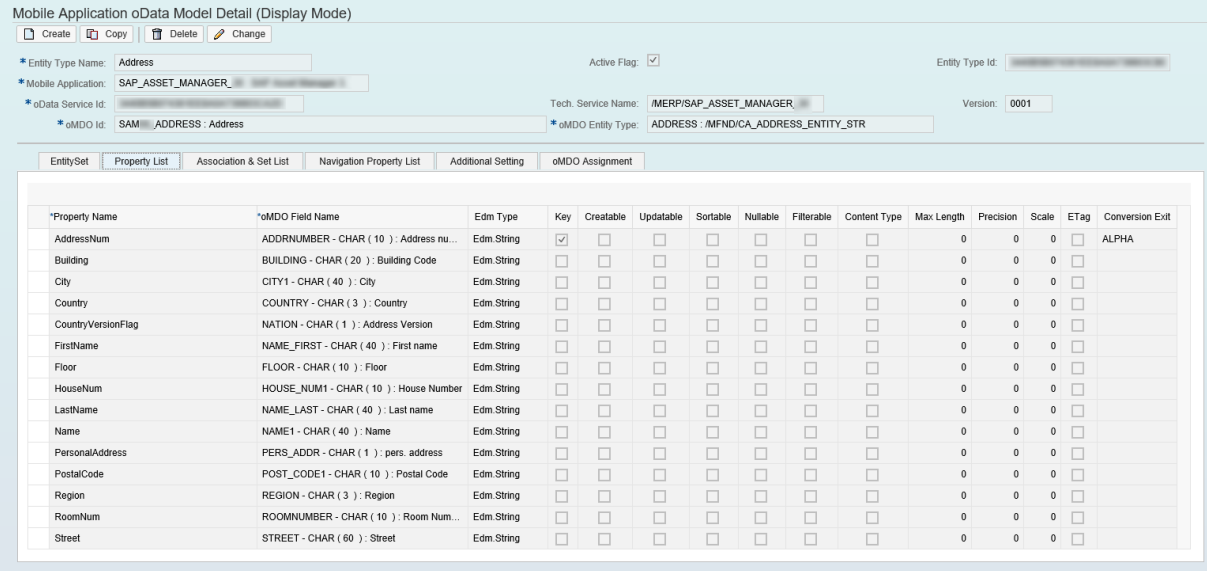
- **Entity Type Name:** Case-sensitive name of the entity type. The name must be unique within the OData service.
- **Active Flag:** If unchecked, the entity type is not included in the generated OData model
- **Entity Type ID:** Internal ID generated by the system to identify the entity type
- **Mobile Application:** Mobile application for the entity type. The OData model configuration is defined for individual mobile applications. You can reuse the entity type name in different mobile applications.
- **Internal OData Service ID:** Internal OData service ID that identifies the OData service for which the entity type is defined
- **Service:** Gateway technical service name of the OData service. Information is read-only.
- **Version:** Gateway technical service version. Information is read-only.
- **OMDO ID:** OMDO that provides business logic for the entity type and its entity set
- **OMDO Entity Type:** Technical entity type of the OMDO that is mapped to the OData entity type. Data for the OData entity type is supplied by the OMDO entity type.

The following attributes are available for the *Entity Set* definition:

- **EntitySet Name:** Case-sensitive name of the entity set. Must be unique within the OData service.
- **Creatable:** If checked, creation (POST) request for the entity set is supported
- **Updatable:** If checked, update (PUT / PATCH / MERGE) request for the entity set is supported
- **Deletable:** If checked, deletion (DELETE) request for the entity set is supported

- **Pageable:** If checked, paging is allowed for the entity set read request
- **Filter Required:** Not applicable for SAP Service and Asset Manager

Property List



*Property Name	*oMDO Field Name	Edm Type	Key	Creatable	Updatable	Sortable	Nullable	Filterable	Content Type	Max Length	Precision	Scale	ETag	Conversion Exit
AddressNum	ADDRNUMBER - CHAR (10) : Address nu...	Edm.String	☑	☐	☐	☐	☐	☐	☐	0	0	0	☐	ALPHA
Building	BUILDING - CHAR (20) : Building Code	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
City	CITY1 - CHAR (40) : City	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
Country	COUNTRY - CHAR (3) : Country	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
CountryVersionFlag	NATION - CHAR (1) : Address Version	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
FirstName	NAME_FIRST - CHAR (40) : First name	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
Floor	FLOOR - CHAR (10) : Floor	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
HouseNum	HOUSE_NUM1 - CHAR (10) : House Number	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
LastName	NAME_LAST - CHAR (40) : Last name	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
Name	NAME1 - CHAR (40) : Name	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
PersonalAddress	PERS_ADDR - CHAR (1) : pers. address	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
PostalCode	POST_CODE1 - CHAR (10) : Postal Code	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
Region	REGION - CHAR (3) : Region	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
RoomNum	ROOMNUMBER - CHAR (10) : Room Num...	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	
Street	STREET - CHAR (60) : Street	Edm.String	☐	☐	☐	☐	☐	☐	☐	0	0	0	☐	

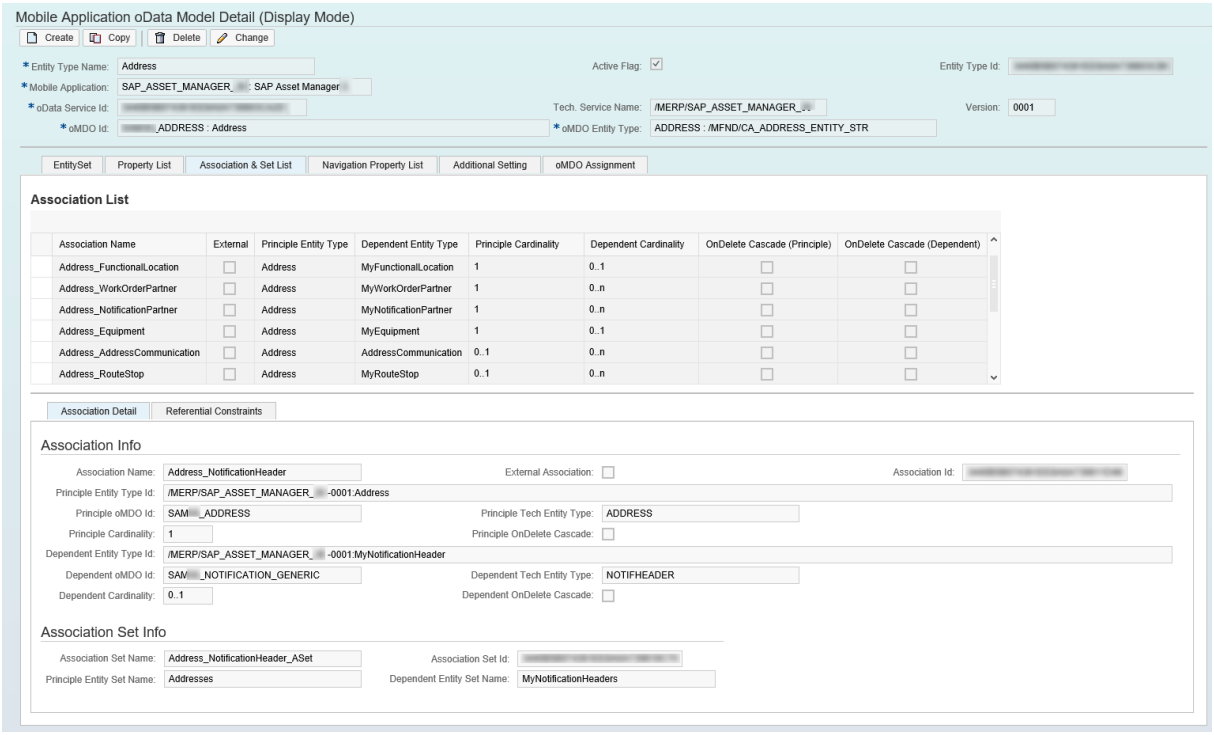
You can configure the following on the *Property List* tab:

- **Property Name:** Case-sensitive name of the property
- **OMDO Field Name:** Field name of the OMDO technical entity type mapped to the property. The value of the property is supplied in the `<OMDO Entity Type>` field.
- **EDM Type:** Standard EDM type for the property
- **Key Indicator:** If checked, the property is the key property of the entity type
- **Creatable:** Not applicable for SAP Service and Asset Manager
- **Updatable:** Not applicable for SAP Service and Asset Manager
- **Sortable:** If checked, the property can be used for sorting
- **Nullable:** If checked, the property can have NULL value
- **Filterable:** If checked, the property can be used for filtering
- **Content Type:** If checked, the property value can represent content type
- **Max Length:** Maximum allowed length of the property value
- **Precision:** Decimal precision of the property, if relevant
- **Scale:** Decimal scale of the property, if relevant
- **ETag:** If checked, the property serves as the `<ETag>` field. You can set only one field as an `<ETag>` field in an entity type.
- **Conversion Exit:** Conversion exit for the property

Association & Set List Tab

An association defines the relationship between two entity types, with one entity type as the principle entity type, and the other as the dependent entity type. An association set defines the relationship between the two entity sets of the respective entity types in the association. In an OData model configuration, associations and association sets are child objects of an entity type, and each association can have only one association set defined.

When you define an OData model to use with OData offline SDK client application, you also define referential constraints for the association. Only key fields of the principle entity type can be used in referential constraints.



You can configure the following in the *Association Info* section:

- **Association Name:** Case-sensitive name of the association
- **External Association:** By default, the parent entity type of the association is the principle entity type. However, if the *External Association* flag is checked, the parent entity type of the association is the dependent entity type. An external association is commonly used when defining associations between entity types across component services.
- **Association ID:** A read-only internal GUID generated by the system that identifies the association
- **Principle Entity Type ID:** Lead entity type of the association
- **Principle OMDO ID:** Read-only OMDO ID to which the principle entity type is mapped to
- **Principle Tech Entity Type:** Read-only technical entity type of the OMDO ID to which the principle entity type is mapped to
- **Principle Cardinality:** Cardinality of the lead entity type
- **OnDelete Cascade (Principle):** If checked, the dependent entity type and entity set are automatically deleted when the principle entity type and entity set are deleted
- **Dependent Entity Type ID:** For standard associations, the dependent entity type can be any entity type belonging to the same OData service or child component service

- **Dependent OMDO ID:** Read-only OMDO ID to which the dependent entity type is mapped to
- **Dependent Tech Entity Type:** Read-only technical entity type of the OMDO ID to which the dependent entity type is mapped to
- **Dependent Cardinality:** Cardinality of the dependent entity type
- **Dependent OnDelete Cascade:** If checked, the principle entity type and entity set are automatically deleted when the dependent entity type and entity set are deleted

You can configure the following in the *Association Set Info* section:

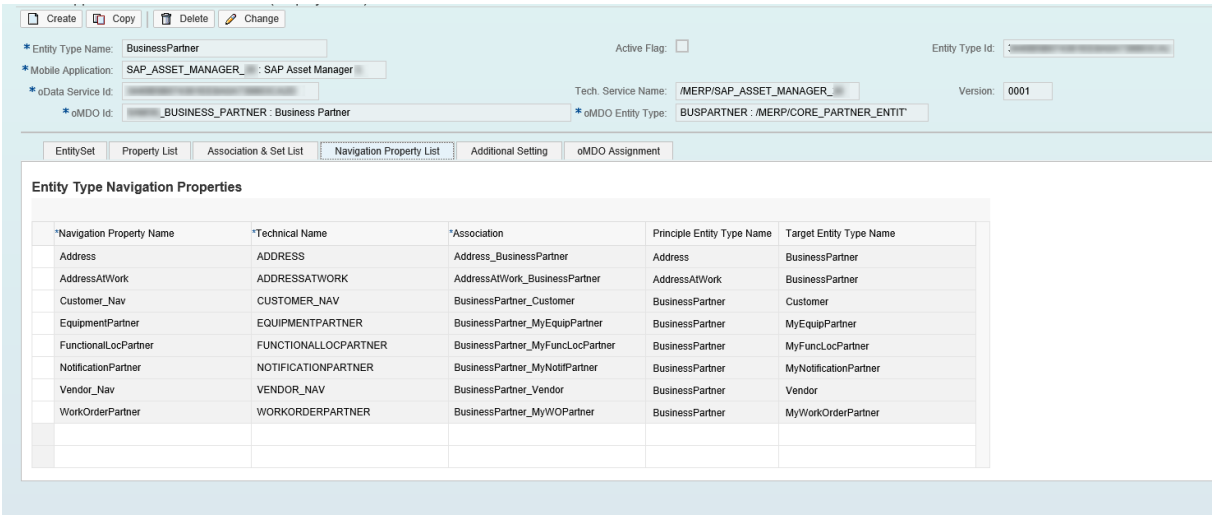
- **Association Set Name:** Case-sensitive name of the association set
- **Association Set ID:** Read-only internal GUID generated by the system that identifies the association set
- **Principle Entity Set Name:** Read-only entity set name of the principle entity type. Each entity type in the OData model configuration can only have one entity set.
- **Dependent Entity Set Name:** Read-only entity set name of the dependent entity type. Each entity type in the OData model configuration can only have one entity set

You can configure the following in the *Referential Constraints* section (not pictured in detail in the example screenshot):

- **Principle Entity Type:** Read-only principle entity type of the association
- **Principle Property:** Principle property of the referential constraint. The principle property is the key field of the principle entity type.
- **Dependent Entity Type:** Read-only dependent entity type of the association
- **Dependent Property:** Dependent property of the referential constraint. The dependent property can be any field of the dependent entity type that has a foreign key relationship with the principle property.

Navigation Property List Tab

A navigation property represents a link from the parent entity type to a related entity types.

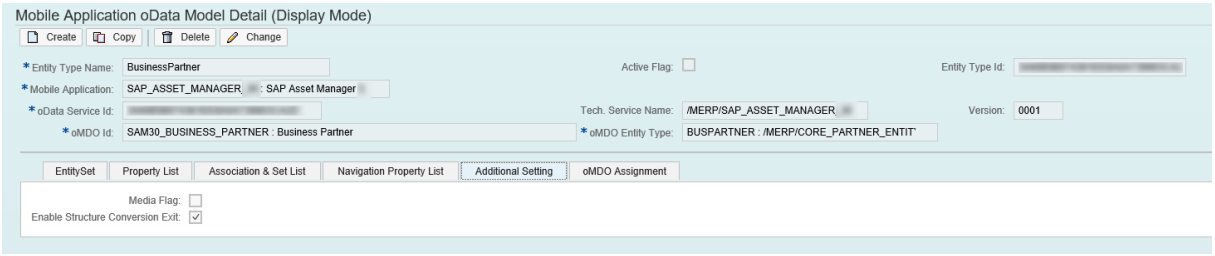


You can define the following attributes for a navigation property in the *Entity Type Navigation Properties* table:

- **Navigation Property Name:**

- **Technical Name:** Case-sensitive name of the Internal technical name of the navigation property. The technical name is not case-sensitive.
- **Association:** Association for the navigation property. The navigation represents the link between the principle entity type and the dependent entity type.
- **Principle Entity Type Name:** Read-only principle entity type name of the association used by the navigation property
- **Dependent Entity Type Name:** Read-only dependent entity type name of the association used by the navigation property

Additional Setting Tab



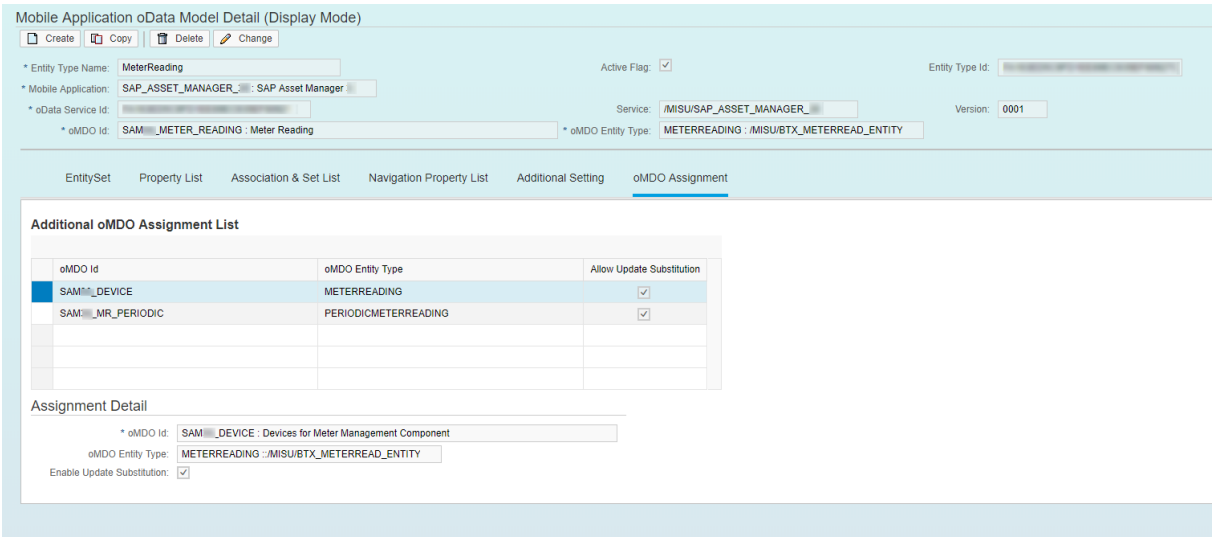
You can define the following additional settings for the OData model:

- **Media Flag:** If checked, the entity type is a media entity type
- **Enable Structure Conversion Exit:** If checked, the gateway automatically performs conversion exits for the entity type at runtime for both the request payload and the response payload

OMDO Assignment Tab

The *OMDO Assignment* tab only supports substitution configuration scenarios.

Use the following screenshot as an example. When a user posts a meter reading from their client, by default the reading is posted to the default OMDO, which here is *SAM<XX>_METER_READING*. However, if the user is reading a periodic meter, the reading is posted to the *SAM<XX>_MR_PERIODIC* OMDO, which is substituted for the default OMDO through the use of custom headers.

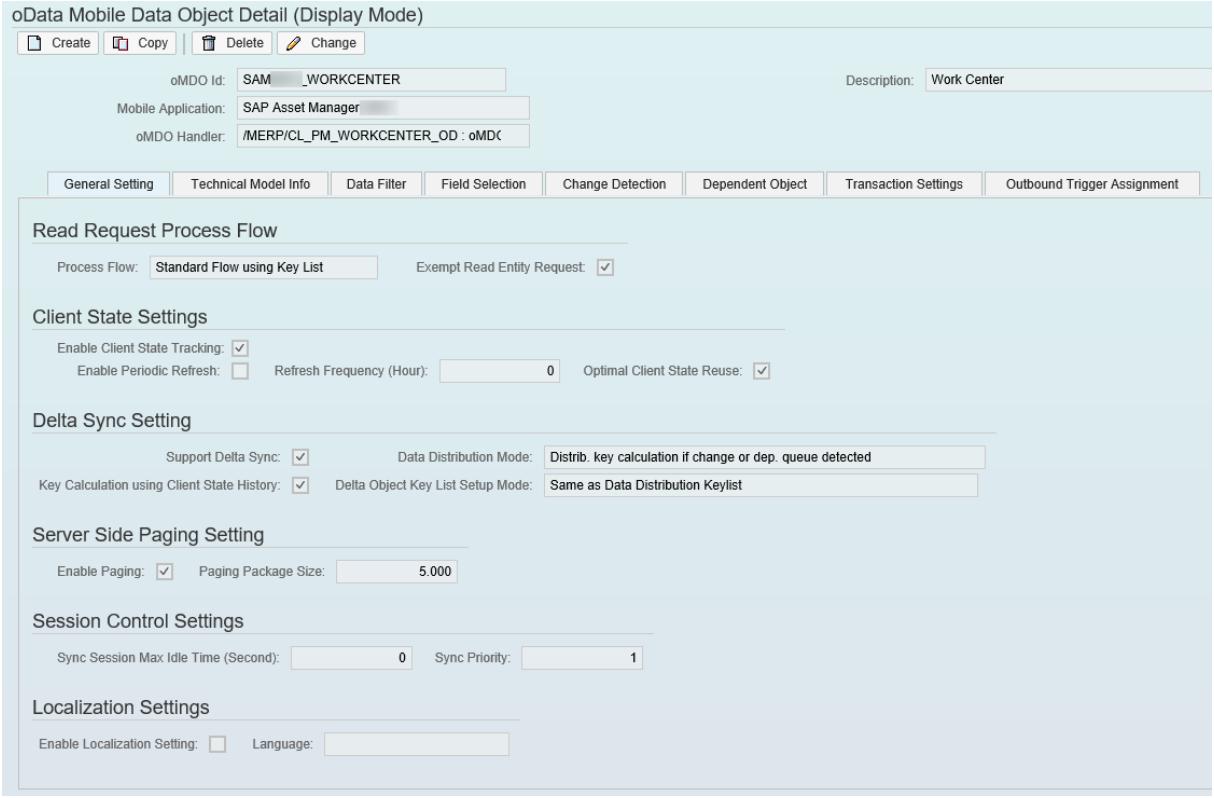


2.2.5.3 OData Mobile Data Object Configuration

An OData mobile data object (also known as OMDO) provides business logic for a business object used in an OData-based mobile application. An OMDO provides both technical implementation and configuration support for the represented business object, including all aspects of related operations such as object creation, update, deletion, or read and downloading. The OMDO also supports configuration such as data distribution rules for data download.

OData requests for a business object are mapped to an OMDO object. The OMDO handler then processes the requests for the OMDO object. For read requests, the OMDO handler considers and enforces the data distribution rules and other configuration settings, and determines the proper output response. For create, update, and delete requests, the OMDO handler creates or updates the SAP BusinessObjects in the back-end system as requested in the OData requests, and provides the relevant response.

General Setting Tab



You can set the following attributes on the *General Setting* tab:

- **OMDO ID:** ID of OData Mobile Data Object; limited to 40 characters. The OMDO ID must be unique in an SAP client, across all mobile applications, as namespace restriction is enforced. A customer-defined OMDO ID must use the Y or Z namespace.
- **Description:** Short description of the OMDO, limited to 60 characters
- **Mobile Application:** Mobile application of the OMDO. An OMDO always belongs to a single mobile application.
- **OMDO Handler:** An ABAP OO class that provides the technical implementation for the OMDO object. The OMDO handler must be a subclass of `/MFND/CL_CORE_OMDO_HNDLER_BASE`. You can reuse an OMDO handler to provide technical implementation for multiple OMDO objects.
- **Process Flow:** Determines how the OMDO handler processes OData entity set read requests. Based on the process flow setting, different OMDO handler methods are invoked at runtime. The OMDO handler determines which process flow it supports.
  - **Standard Flow using Key List:** Typically used by the OMDO handler to support complex SAP BusinessObjects with multiple entity types, complex relationships between entity types and data distribution rules, and with change detection support. Examples of SAP BusinessObjects that use the standard flow include transaction data objects such as work orders or notifications; and leading data objects such as equipment or functional locations.
  - **Basic Flow without Key List:** Typically used by the OMDO handler to support simple SAP BusinessObjects with a single entity type and no change detection support. Examples of SAP BusinessObjects that use the basic flow without key list include customizing data, such as plant

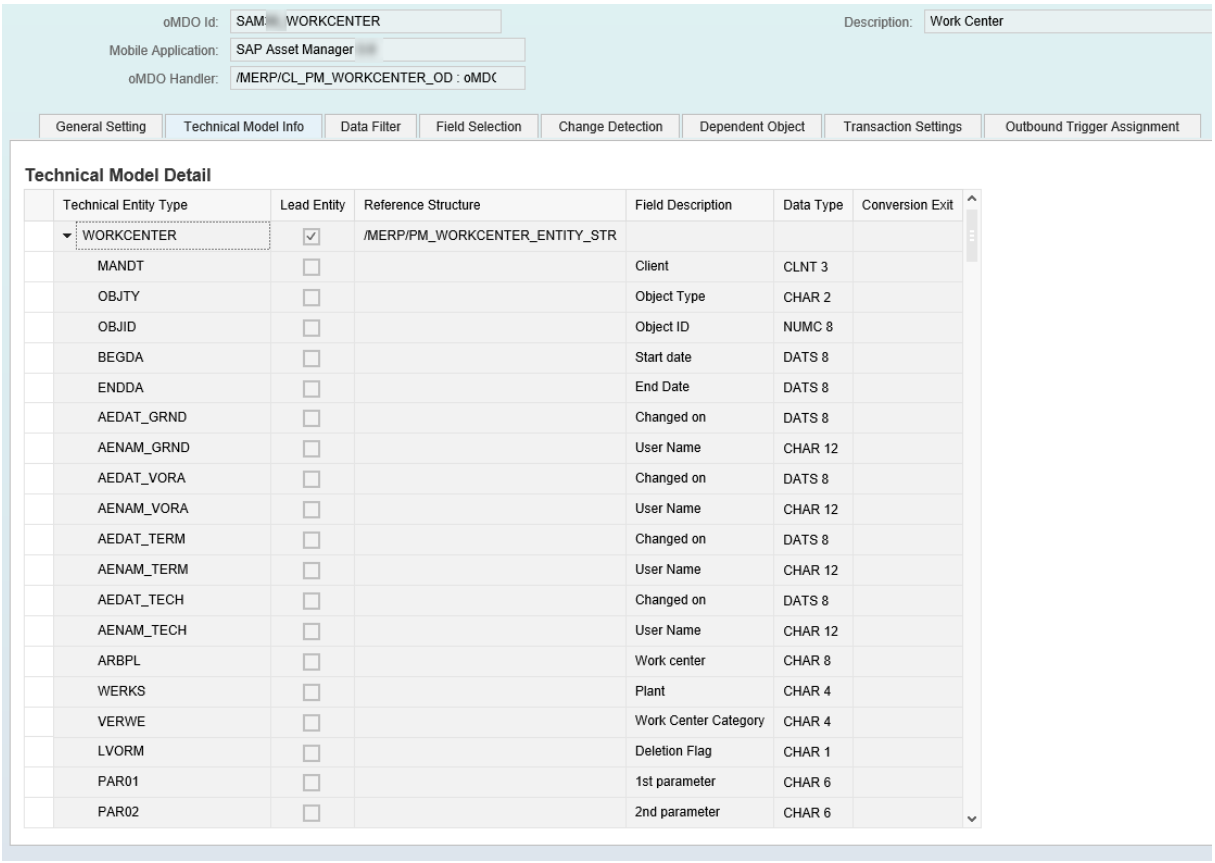
or order type. OMDO handlers that only support basic flow are typically subclasses from `/MFND/CL_CORE_OMDO_BASIC_HNDLR`.

- **Exempt Read Entity Request:** If checked, the entity read request is handled by the *READ_ENTITY_REQUEST_PROC* method. The method supports on-demand entity read requests without interference with client state tracking for entity set read requests.
- **Enable Client State Tracking:** If checked, the client state records are maintained for each entity set read request. Client state tracking enablement is required for other features such as key list calculation using client state history, periodic refresh support, and optimized client state reuse.
  - **Client State:** Captures the list of calculated object keys that are sent to the mobile client for the entity set read requests, the data distribution rules used for the calculation, and the time of the calculation. By enabling client state management, the system has a record of the objects distributed to the mobile client. Enabling client state management also allows the system to calculate the list of objects to be removed from mobile client via tombstones.
    At runtime, the client state info generated during client synchronization can be displayed using the client state monitor through the Administration & Monitoring Portal.
- **Enable Periodic Refresh:** If checked, every entity set delta sync read request is checked if it qualifies for periodic refresh. A periodic refresh means that if time between an entity set initial sync or last period refresh read request and current read request has exceeded the defined frequency, all objects from the client are removed and a new object key list is calculated based on the current data distribution rules.
- **Refresh Frequency (Hour):** Defines the periodic frequency refresh in number of hours
- **Optimal Client State Reuse:** If checked, the system tries to reuse previous client state records whenever possible instead of generating new client state records. Reuse of previous client state records can improve runtime performance. Whether a client state record can be reused depends on the object key list and data distribution rules. If both items remain unchanged compared to the previous client state, the client state record can be reused.
- **Support Delta Sync:** If checked, a delta token is generated and returned in response to the entity set read request
- **Key Calculation using Client State History:** This setting is effective only if client state tracking is enabled. If checked, the object key list is taken from current data distribution calculation, and the list is compared against the object key list from the previous client state. Also, objects that should be removed from the client are automatically identified, and downloaded to the client again.
- **Data Distribution Mode:** Controls when data distribution key calculation is performed for delta sync entity set read requests. It's only relevant if OMDO handler supports the *Standard Flow using Key List* process flow. There's overhead associated with data distribution key calculation. You can reduce the number of key calculations during sync to improve performance. However, the potential performance gain has to be measured against the correctness of the response. If the object key list from data distribution changes frequently, skipping the calculation can lead to incorrect results.
  - **Always perform distribution key calculation:** Perform the data distribution key calculation for every read request
  - **Distribution key calculation if change or dependent queue detected:** Perform data distribution key calculation only if there's data change detected in the back-end system, or if there are new dependent queue entries inserted
  - **No distribution key calculation. Delta calculation only:** Always skip data distribution key calculation in delta sync
- **Delta Object Key List Set up Mode:** Used in standard flow to determine how the delta object key list is initialized. The delta object key list determines the list of objects that are included in the delta sync response.

    - **Same as Data Distribution Key List:** Default delta sync object key list is the same as the data distribution key list. If change detection is supported by the OMDO, we recommend using this setting. Remove objects that do not have changes since the last delta sync from the delta object key list to prevent unnecessary download and to improve performance.
    - **Difference of Current and Last Data Distribution Key List:** The current data distribution key list is compared to the object key list from the last client state. The only objects that are not included in the last client state object list are included in the delta sync object key list. If no change detection is supported by the OMDO, we recommend using this setting.
- **Enable Paging:** If checked, server-side paging is activated. When paging is active, response from the OMDO to the OData request is limited to the specified paging package size. If the number of records in the response exceeds the paging package size, the response is divided into separate pages. The *$skiptoken* identifies the next page that is generated and included in the response to the client. The client then sends a follow-up request with the skip token to retrieve the next page. This process continues until the client retrieves all pages.
- **Paging Package Size:** Maximum number of records that can be included in a page for the response of the OMDO. For read requests in a batch, the maximum payload records number defined for the OData service assignment is also considered. If the overall number of records in the batch response has reached the maximum payload records number, the number of records in the individual request response can be less than the paging package size specified.
- **Sync Session Max Idle Time (Second):** Estimated maximum duration of a sync session for a single OMDO. For requests belonging to the same OMDO, the response can be calculated beforehand for all requests once and then reused. When the requests are received separately, as long as the requests are received within the defined sync time duration, the response calculated beforehand can be used, instead of calculating it again. Calculating a response beforehand can improve performance. If the sync session max idle time is set to *0*, each request triggers the calculation for the response.
- **Sync Priority:** Represents the processing sequence for read requests in a batch, where *0* has the first priority sequence. For OMDO objects with the same sync priority, OData read requests for these OMDO objects are processed based on the sequence in the batch.
- **Enable Localization Setting:** When checked, brings data associated with the OMDO fetch translated to the language specified in the `<Language>` field irrespective of what logon language is used for the connection. All other OMDOs that don't have *Enable Localization Setting* enabled use the logon language.
- **Language:** Used when the `<Enable Localization Setting>` box is checked. Select your desired language from the dropdown list.

Technical Model Info Tab

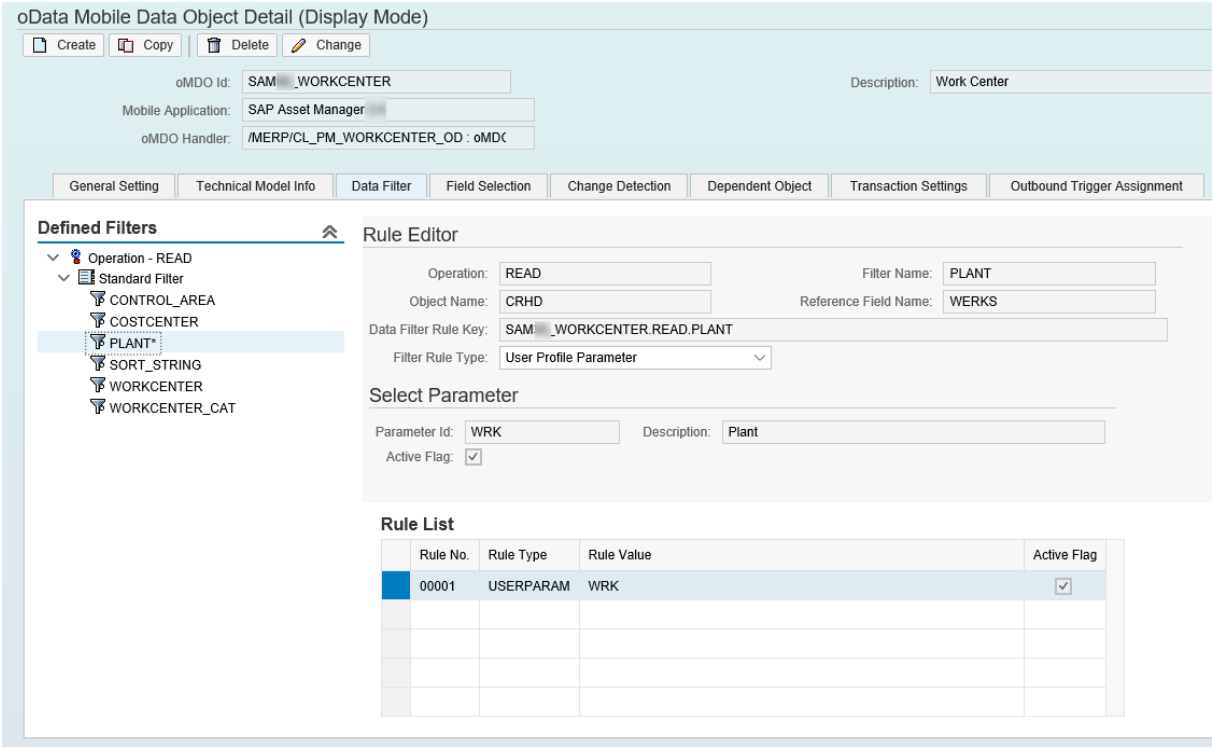
The *Technical Model Info* tab is a display only tab. This tab displays the technical entity model supported by the OMDO handler.



- **Technical Entity Type:** Technical entity type that the OMDO handler supports
- **Lead Entity:** Indicates whether a technical entity is the lead entity type supported by the OMDO handler. The lead entity type represents the header record of a business object. An OMDO operates on a business object level. For an OMDO CREATE operation, a create request (POST request) for the lead entity type is required. If the lead entity already exists, a CREATE request (POST request) for nonlead entity types are considered as OMDO UPDATE operations.
- **Reference Structure:** Data dictionary structure of the technical entity type
- **Field Name:** Field name from the data dictionary structure
- **Field Description:** Field description
- **Data Type:** Field data type
- **Conversion Exit:** Assigned conversion exit for the field

Data Filter Tab

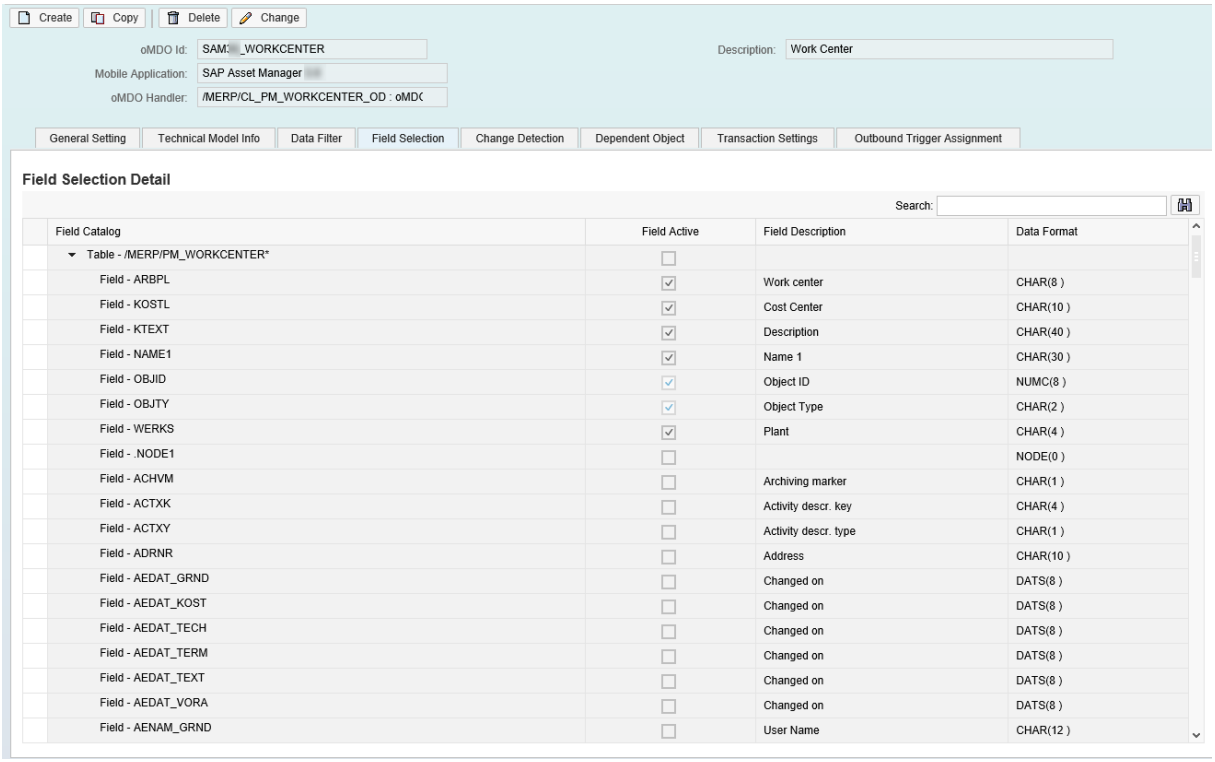
An OMDO handler can declare data filters and parameters supported by its CRUD (CREATE / READ / UPDATE / DELETE) operations. These filters are displayed on the *Data Filter* tab.



- **Defined Filters:** The *Defined Filters* navigation area displays the list of supported filters for the OMDO, grouped by operation and filter group. There are two types of filters available, though both types may not be available for every OMDO:
  - **Standard Filter:** A standard filter is a single field filter that references a table or structure field that is displayed in the `<Object Name>` field and the `<Reference Field Name>` field
  - **Table Filter:** A table filter is a structured filter that references a table or structure that is displayed in the `<Object Name>` field.
- **Rule Editor:** Details for a filter are displayed in this section when you select a filter from the *Defined Filters* section.
- **Filter Rule Type:** You can select one of the four types of rules to define:
  - **Static Value in Range Table Format:** Static rule, with the rule value defined at design time in the configuration
  - **User Profile Parameter:** Dynamic rule, with the rule value evaluated at runtime based on the runtime profile parameter value of the user. You can display and maintain the user profile parameter value with the transaction code *SU3*.
  - **Mobile User Attribute:** Dynamic rule, with the rule value evaluated at runtime based on the runtime mobile user attribute of the user. You can display and maintain the mobile user attribute in the Administration & Monitoring Portal, which you can access with the transaction */SYCLO/ADMIN*. Then select *Administration* > *User Management*.
  - **Filter Handler:** Dynamic rule, with the rule value evaluated at runtime by a filter handler. A filter handler is an ABAP subclass of `/MFND/CL_CORE_OMDO_RULE_BASE`.

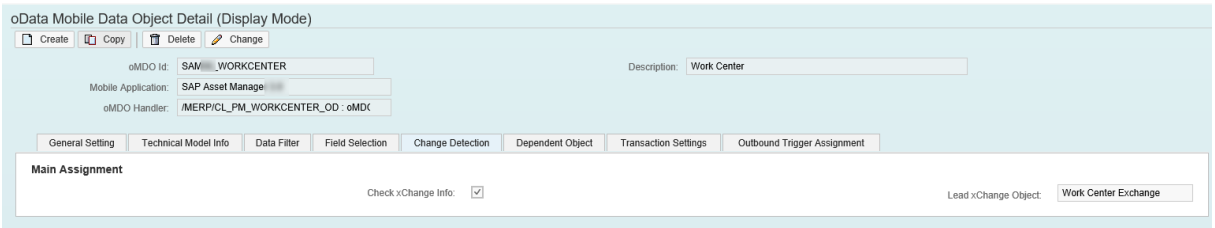
Field Selection Tab

An OMDO handler can declare field catalogs supported for the READ operation. If there is a READ operation, by default, all of the fields from the database tables related to the OMDO object are selected. Using the field catalog, customers can control which fields are selected, and improve performance, as typically a mobile application doesn't require all of the fields.



Change Detection Tab

You can enable change detection for the OMDO using the *Change Detection* tab.



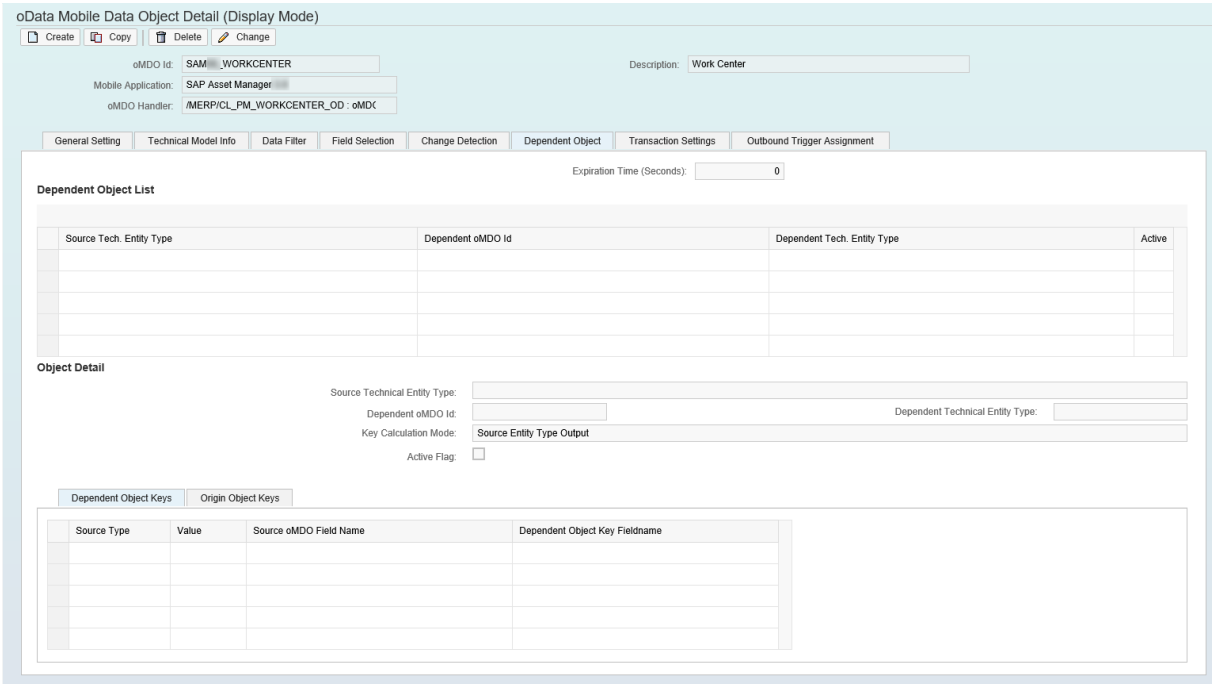
- **Check xChange Info:** Applies to standard flow processing only. If checked, change detection info is checked to determine the delta sync object key list.
- **Lead xChange Object:** xChange object that supplies the change detection information for the OMDO. Information from the xChange table of the xChange object is read and used for the calculation of the delta sync object key list.

Dependent Object Tab

Define settings related to dependent objects on the *Dependent Objects* tab.

In some business cases, the read request sequence for the OMDOs or SAP BusinessObjects is important, since the data distribution object key list of a subsequent OMDO depends on the results or outputs of the precedent OMDOs. The subsequent OMDO is treated as a dependent object of the precedent OMDO. The leading OMDO is the source OMDO, as the output of the lead OMDO supplies information for the dependent OMDO. Dependent object key information generated by the leading OMDO is stored in the dependent object queue, and is used by the dependent OMDO during its read request processing.

For example, SAP Service and Asset Manager downloads detailed information for equipment and functional locations used in work orders assigned to a technician. To fulfill this requirement, read requests for work order assignments occur first, and equipment and functional locations are set up as dependent objects for the work order OMDO.



You can define the following settings for a dependent object of the current OMDO:

- **Source Technical Entity Type:** Source OMDO technical entity type that contains information required by the dependent object
- **Dependent OMDO ID:** ID of the dependent OMDO
- **Dependent Technical Entity Type:** Receiving technical entity type of the dependent OMDO, for which information from the source technical entity type is transferred
- **Key Calculation Mode:** Select the way the keys are passed to the OMDO. Key calculation is a dependent object concept; how you set up your dependent object is based on your source object.
  - Source Entity Output: Input for the dependent key. Keys are calculated based on the source entity type output.
  - Source Entity Type Distribution Key List: Dependent Object Key construction comes from the distribution key list of the source entity type. Using this option always collects all the valid keys from the source entity type.

    - Source Entity Type Output + Target Entity Type Client State: Similar to Source Entity Output plus the previous client state of the target entity type. Here, what is being created for dependent object collection is a combined collection of the source entity type output and the target entity type client state records from the previous sync.
- **Active Flag:** Enable or disable a dependent object definition

You can define the following settings for the mapping info of dependent object keys in the *Dependent Object Keys* tab:

- **Source Type:** Use option *By Field Name* if the information comes from a field of the source technical entity type. Use option *By Value* if a constant value is used.
- **Source Value:** Constant value for a dependent object key field. This field is only relevant if the source type is set to *By Value*.
- **Source OMDO Field Name:** Name of the source technical entity type field that supplies value for the dependent object key. This field is only relevant if the source type is set to *By Field Name*.
- **Dependent Object Key Field Name:** Field name of the dependent technical entity type that receives the value from the source technical entity type field

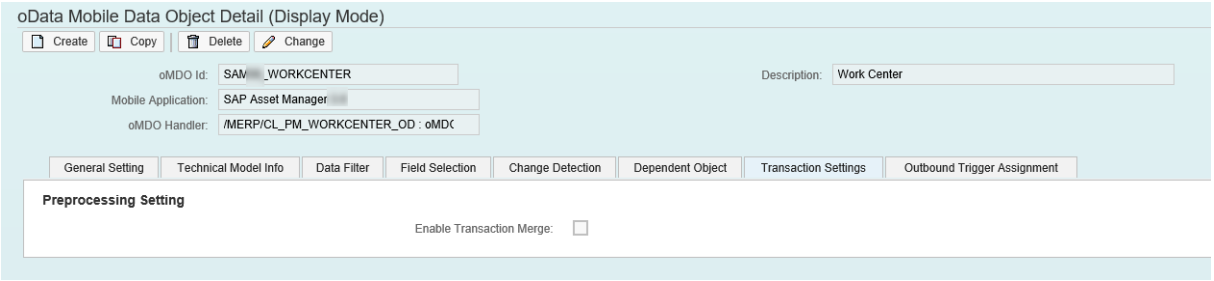
You can define the following settings for the mapping info of origin object keys in the *Origin Object Keys* tab (not shown in detail in the example screenshot). The origin object key identifies the source OMDO object that has generated the dependent object key.

- **Source Type:** Use option *By Field Name* if the information comes from a field of the source technical entity type. Use option *By Value* if a constant value is used.
- **Source Value:** Constant value for an origin object key field. This field is only relevant if the source type is set to *By Value*.
- **Source OMDO Field Name:** Name of the source technical entity type field that supplies value for the origin object key. This field is only relevant if the source type is set to *By Field Name*.

You can display the dependent object queues generated during client synchronization at runtime using the *Dependent Queue Monitor* on the Administration & Monitoring Portal.

Transaction Settings Tab

You can define settings related to transactions (CUD requests) on the *Transaction Settings* tab.



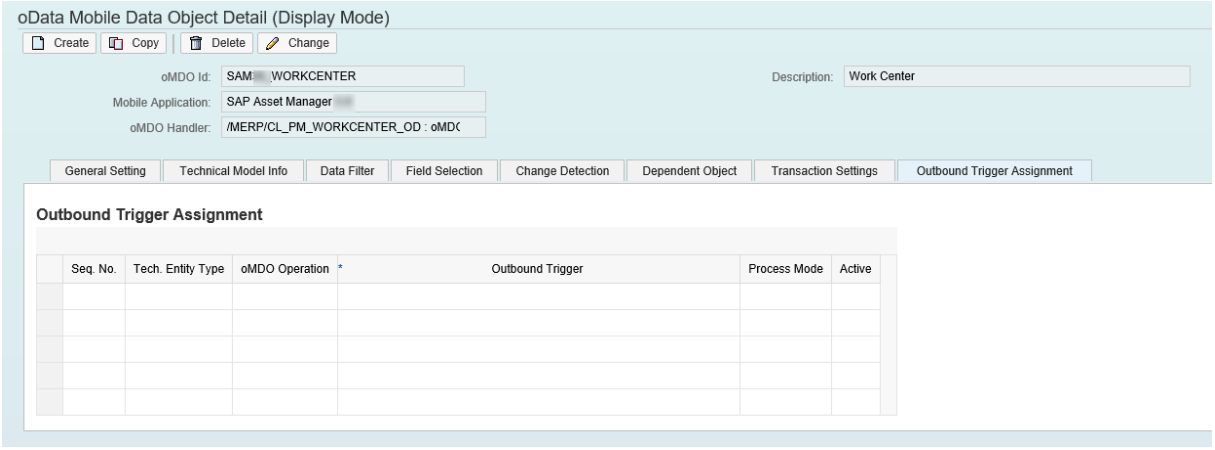
- **Enable Transaction Merge:** If checked, transaction requests for the same object that are received in the same changeset are merged. Therefore, the number of requests processed by the OMDO handler is reduced. The sequence of the transaction requests in the changeset is respected, with the attribute value of the last transaction request as the final value for the attribute.

For example, for *Object 123* the requests are as follows:

Requests in Changeset	Requests Processed by OMDO
Request #1 CREATE 123 Request #2 UPDATE 123 Request #3 DELETE 123	None
Request #1 CREATE 123 Request #2 UPDATE 123 Request #3 UPDATE 123	Request #1 CREATE 123 (attribute values from Request #2 and Request #3 are merged into Request #1)
Request #1 UPDATE 123 Request #2 UPDATE 123	Request #1 UPDATE 123 (attribute values from Request #3 merged into Request #1)

Outbound Trigger Assignment

An outbound trigger performs a function that is implemented by the outbound trigger handler. Outbound triggers can be assigned to an OMDO. The assigned outbound triggers are invoked after OMDO processing has been completed, based on the sequence of the assignment.



You can set the following attributes when assigning an outbound trigger to an OMDO:

- **Technical Entity Type:** Optional. If defined, the outbound trigger is invoked only if the specified technical entity type was processed by the OMDO.
- **OMDO Operation:** Optional. If defined, the outbound trigger is invoked only if the specified OMDO operation is processed.
- **Outbound Trigger ID:** Assigned outbound trigger ID
- **Process Mode:** Only the *Always Run* mode is supported
- **Active:** Enable or disable an outbound trigger

2.2.6 Change Detection Settings

Change detection settings are used to define and configure how the mobile application, such as SAP Service and Asset Manager, communicates with SAP and the object tables contained within SAP

The following areas are used to configure backend change detection:

- **Exchange Object Configuration:** Change detection rules for SAP data objects, such as leading data and transaction data, defined for each mobile application
- **EFI Assignment:** Enhancement framework implementation trigger assigned to exchange objects

> ⓘ Note
>
> Create tables and objects in SAP and the Mobile Development Kit before you can create or configure information in the ConfigPanel.

2.2.6.1 EFI Assignment

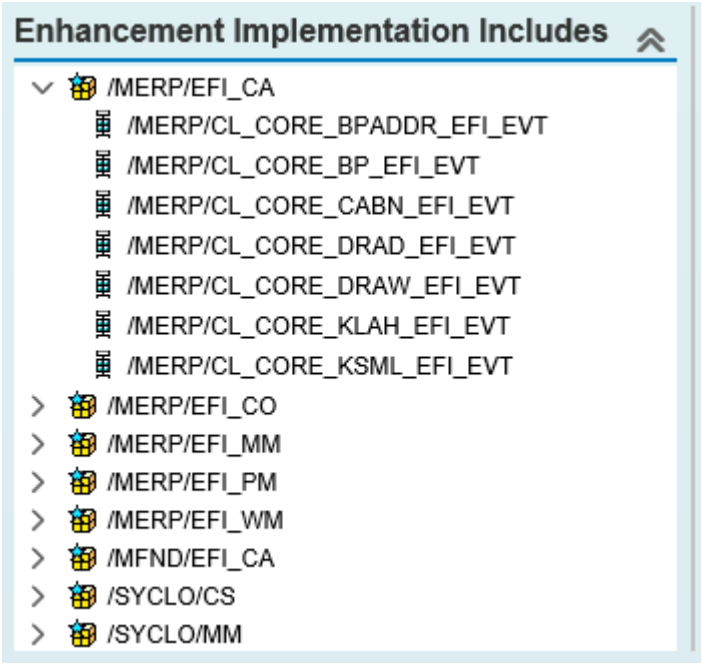
Enhancement Framework Implementation (EFI) source code plug-ins are implemented by the SAP Mobile Add-On for each business object where you configure change detection.

The source code plug-in is provided as an ABAP include file. Each exchange object is assigned to a plug-in to handle the actual change detection process. EFIs are typically available across multiple mobile applications running on the same system.

EFIs collect before and after images of data in an SAP object that was created, modified, or deleted. The EFI then hands those images to the exchange object, which continues with the data processing. Therefore, link the EFIs to their corresponding exchange objects.

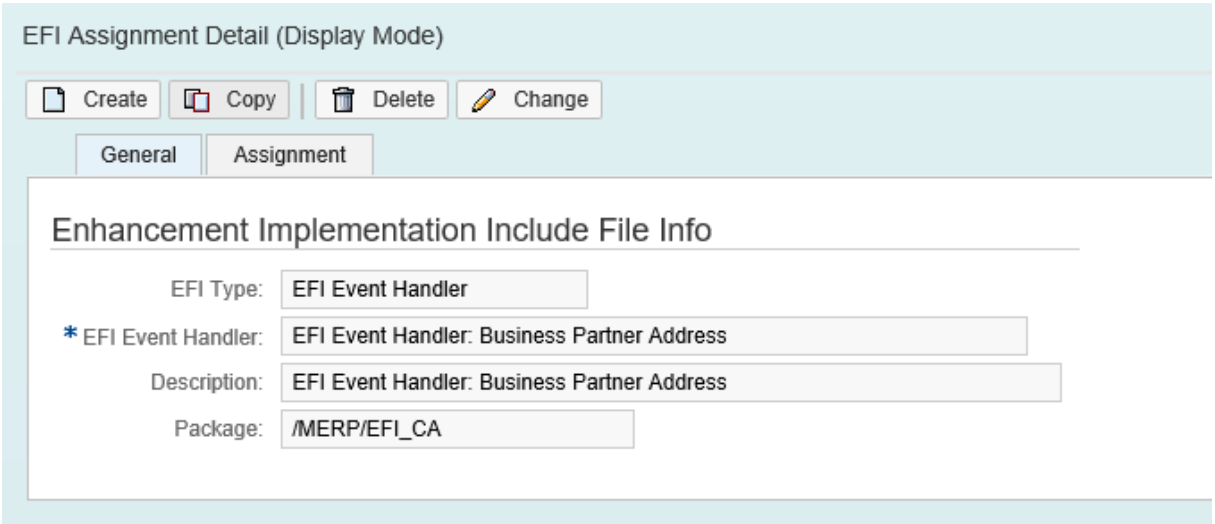
Enhancement Implementation Includes Section

The Enhancement Implementation Includes section is a tree of the include file list in the package. To expand the list, click the arrow to the right of the first item.



General Tab

Use the *General* tab to view and modify the general settings for chosen EFI file.

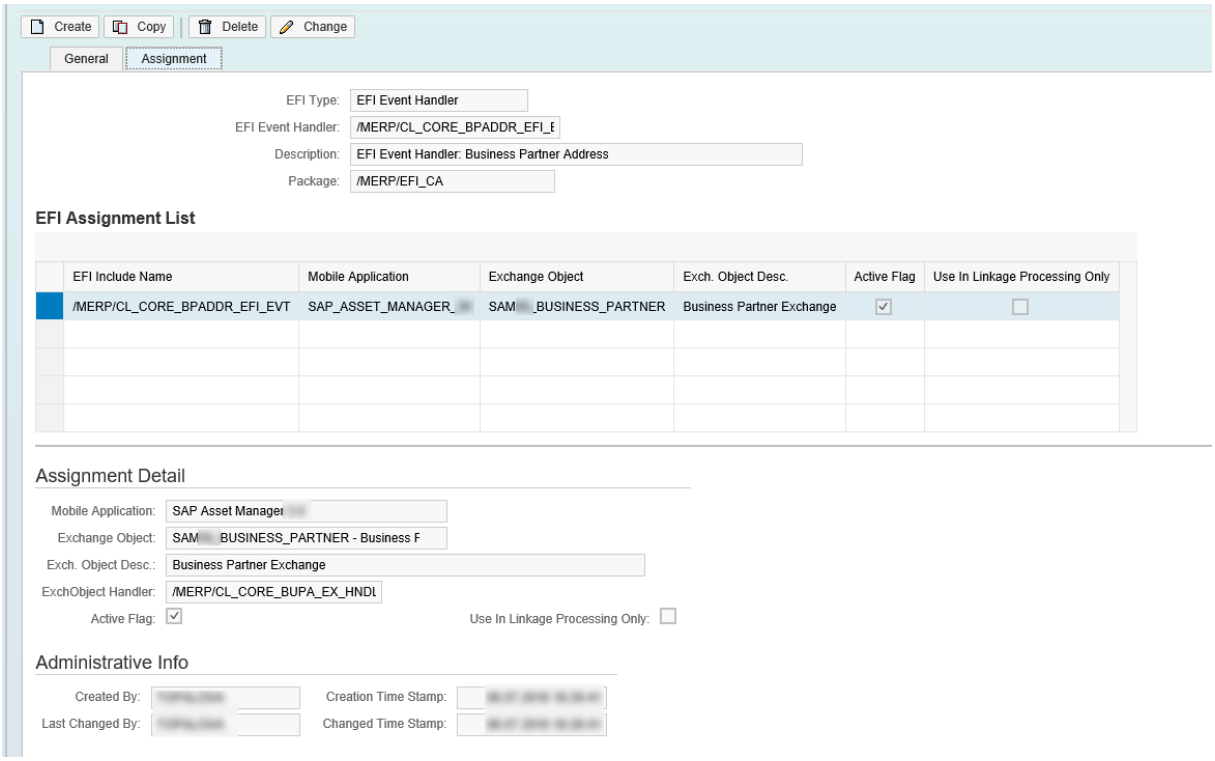


- **EFI Type:** Select one of two options; *Standard EFI Include* or *EFI Event Handler*. Choosing Standard EFI Include is the traditional way to implement EFI and configure the EFI assignments. Selecting EFI Event Handler implements EFI using an ABAP class-based approach.
  When you use a class-based approach, EFI implementation is developed as a subclass of `/SMFND/CL_CORE_EFI_EVENT_BASE`. Available EFI event handler classes are displayed in the dropdown field. The EFI class-based approach provides a more robust functionality and is recommended for a new EFI implementation.

- **EFI Include Name:** File name of the source code plug-in
- **Description:** Short description of the EFI. The description field is automatically populated when you select the EFI include name and is read only.
- **Package:** Package where the EFI is located. The package field is automatically populated when you select the EFI include name and is read only.

Assignment Tab

Use the *Assignment* tab to modify the EFI assignments.



- **EFI Information fields:** The EFI information fields at the top of the *Assignment* tab, like `<EFI Type>` and `<EFI Event Handler>`, are taken from information in the *General* tab and are read only.
- **EFI Assignment List:** Table that displays the plug-ins that are assigned to a specific include file. All column information is replicated in the *Assignment Detail* section directly below the table.
- **Mobile Application:** Read-only name of the specific mobile application
- **Exchange Object:** Name of the exchange object to which the EFI include file is assigned
- **Exchange Object Description:** Read-only description of the exchange object
- **Exchange Object Handler:** Read-only name of the class handler from the repository responsible for updating the exchange table
- **Active Flag:** When checked, the exchange object is in an active state. If unchecked, the EFI isn't linked to the assigned OMDO.
- **Use in Linkage Processing Only:** When checked, the xChange object is only allowed during linkage processing. If not checked, the original EFI is triggered during xChange processing.

2.2.6.2 Exchange Object Configuration

The exchange object defines what in the exchange table is updated in the exchange persistent layer, what class handler is called to update the exchange table, and what fields are related to the change detection.

Use the Configuration Panel to specify which changes are relevant to the mobile application and what conditions to satisfy for so that an update action is triggered. The *Exchange Object Configuration* panel has the following tabs:

- Technical Settings
- Change Detection Field Selection
- Change Detection Condition Filter
- Data Segment Settings
- Linkage Settings
- Push Settings

Technical Settings Tab

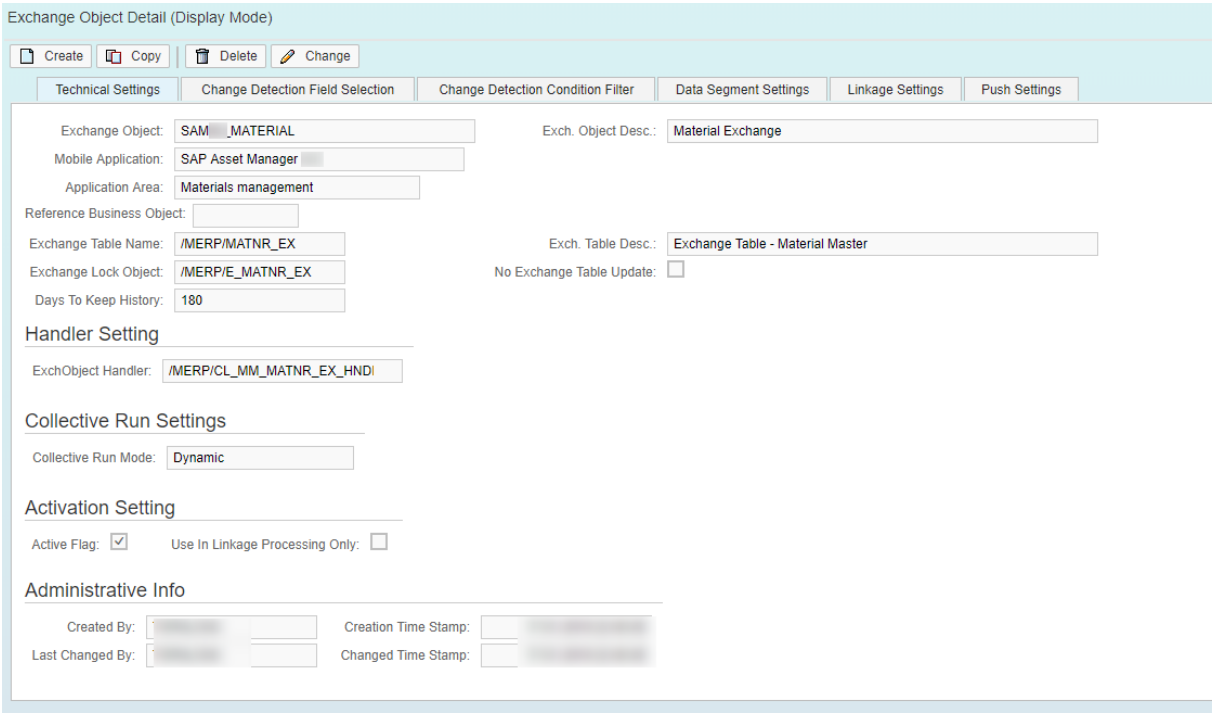
Use the Technical Settings tab to configure basic settings for an exchange object.

Use the `<Exchange Object>` field for the ID of the exchange object, limited to 40 characters. Type in a description in the `<Exchange Object Description>` field, limited to 60 characters. The `<Mobile Application>` field contains a dropdown where you can select your mobile application. The `<Application Area>` classifies the exchange object based on standard SAP application areas using a dropdown selection field.

The `<Reference Business Object>` is the standard SAP business object. The `<Exchange Table Name>` is the name of the table stored in SAP that contains the technical data. The `<Exchange Table Description>` is a brief description of the exchange table. The `<Exchange Lock Object>` field is used when updating the exchange table. Type in how many days you want to keep historical data in the `<Days to Keep History>` field. Check the `<No Exchange Table Update>` checkbox to not write the record to the exchange table in SAP when the record is changed.

- **Handler Setting:** Type in the name of the class handler from the repository that is responsible for updating the exchange table in the `<Exchange Object Handler>` field.
- **Collective Run Settings:** Define the condition where xChange processing is executed asynchronously as a V3 run by selecting one of the following mode options:
  - Dynamic: The collective run mode is determined at runtime by the xChange handler method `DETERMINE_EXEC_MODE`
  - Not Allowed: Not allowed to switch to collective run mode
  - Activated: Always execute asynchronously in V3 collective run mode
  - By User Parameter ID: Switch to V3 collective run mode for runtime user with the specified user parameter value set in the user profile
- **Activation Setting:** Check the `<Active Flag>` checkbox to ensure that the exchange object is in an active state. If unchecked, the exchange object performs no actions. When the `<Use in Linkage Processing Only>` checkbox is checked, the xChange object is only allowed during linkage processing and not if the original EFI was triggered during the xChange process.

The following screenshot shows an exchange process enabled for *MATERIAL*. Any changes for the MATERIAL leading data are recorded in the exchange table and are transmitted to the client during the next transmit.



Change Detection Field Selection Tab

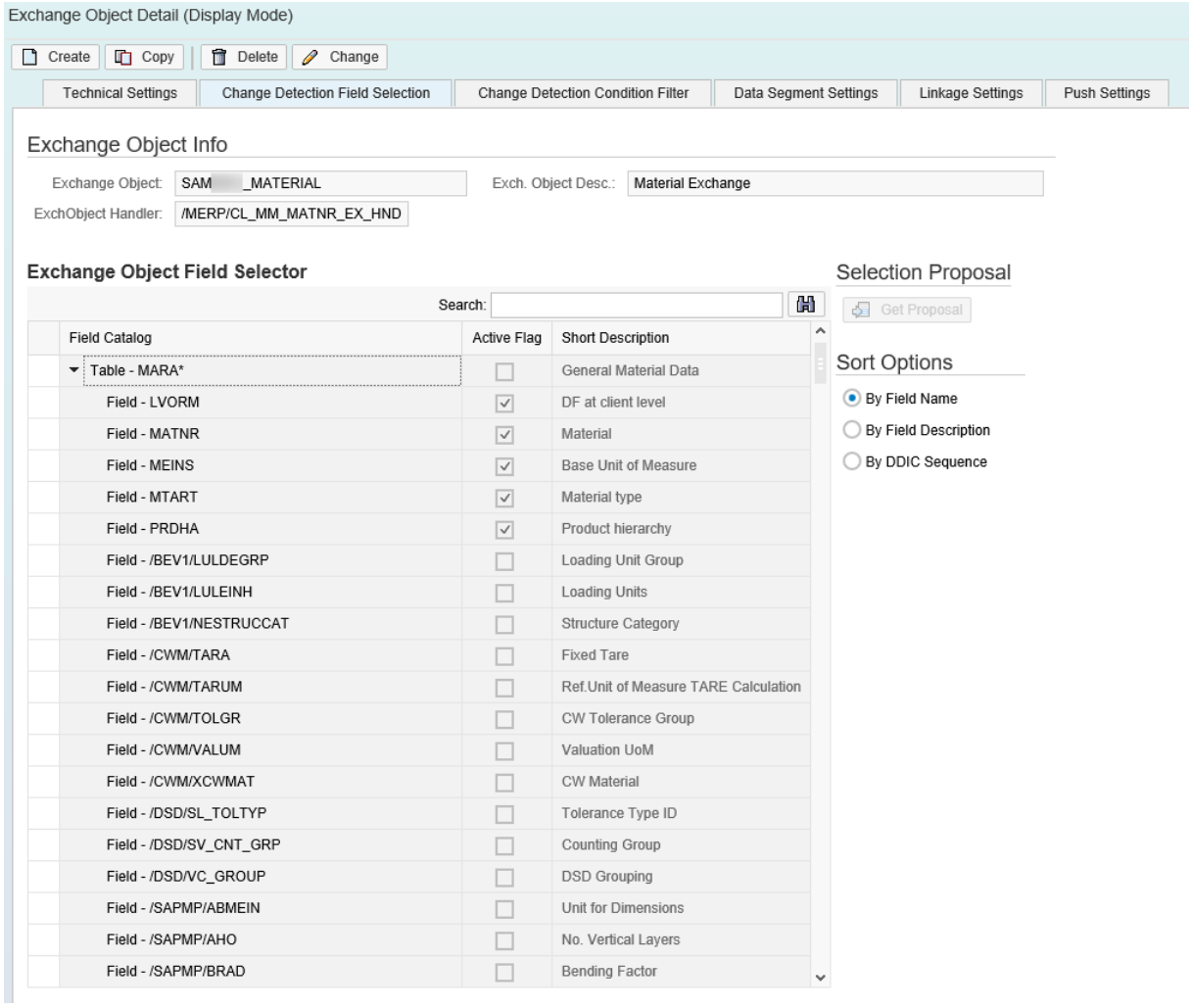
The Change Detection Field Selection tab lets you optimize the change detection process for a mobile application. If a value change is detected for any fields within the group, the object identifier is written to the exchange table, indicating that a change was made. If the `<Active Flag>` is not checked for a field, any value changes made to that field are not detected and recorded to SAP during the exchange process. By default, all fields are initially checked.

The *Exchange Object by Application* tree lists all application areas and the exchange objects linked to each application area. Expand the tree by clicking on the arrows to the right of the application area to display the exchange objects associated with it.

- **Exchange Object Info:** The `<Exchange Object>` field is read only and is the ID of the exchange object. The `<Exchange Object Description>` is read only and is a brief description of the exchange object. The `<Exchange Object Handler>` field is read only and is the name of the class handler from the repository that is responsible for updating the exchange table.
- **Exchange Object Field Selector:** The `<Field Catalog>` column is comprised of non-editable rows of all fields that are detected by the class handler when changes are made. These fields are grouped by the technical table name of the SAP business object.
  When the `<Active Flag>` checkbox is checked, either the table or a field within the table is active. Any value change to the selected field is detected by the class handler. Note that if you check the Active Flag checkbox on a table row, it selects all the rows within the table.
  The `<Short Description>` is a read only field that contains a brief description of the table or of a field withint the table.

- **Selection Proposal:** In a typical mobile application installation, you do not want to have all fields marked as active for change detection. Rather, only the fields that are active on the odata mobile data object that are brought down to the mobile device will also be active in the exchange object. Based on odata mobile data object usage in the application, the selection proposal examines the active flags that are checked for an exchange object's table fields and provides recommendations to the administrator on which fields should be checked or unchecked.

See the following screenshot for an example of the enabled exchange object *MATERIAL*, where the properties of the object are captured and recorded in the exchange table. The properties that trigger the exchange are defined on this *Change Detection Condition Filter* tab, as seen in the checked `<Active Flags>`:

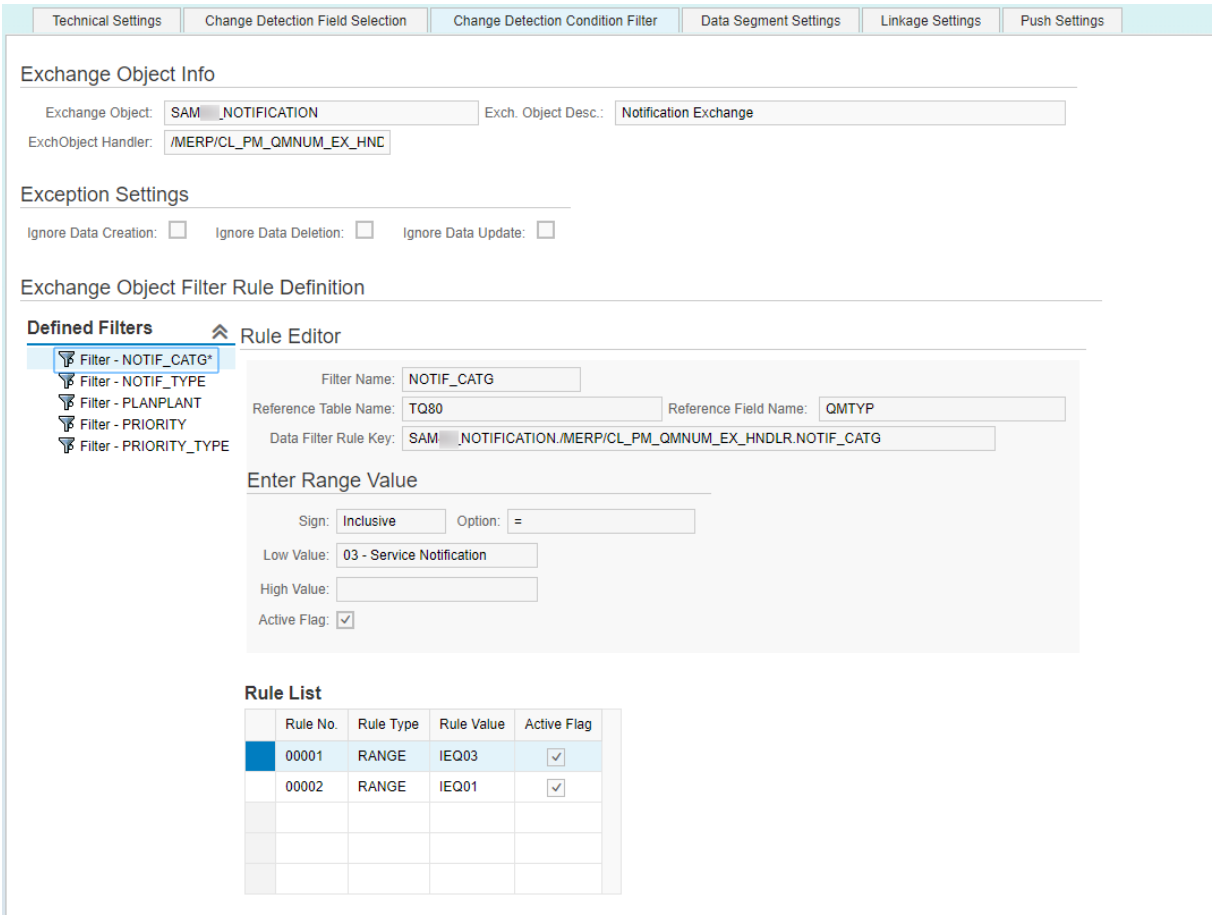


Change Detection Condition Filter Tab

The Change Detection Condition Filter tab lets you restrict change detection based on data content. For exchange handlers to support this feature, define data filter conditions for which the underlying SAP business object must qualify before the change detection process is triggered. The condition is defined at the table field level and is in the SAP range table format.

- **Exchange Object Info:** The `<Exchange Object>` field is read only and is the ID of the exchange object. The `<Exchange Object Description>` is read only and is a brief description of the exchange object. The `<Exchange Object Handler>` field is read only and is the name of the class handler from the repository that is responsible for updating the exchange table.
- **Exception Settings:** When the `<Ignore Data Creation>` checkbox is checked, newly created records and data are not processed to the exchange table. When the `<Ignore Data Deletion>` checkbox is checked, deleted records and data are not processed to the exchange table. When the `<Ignore Data Update>` checkbox is checked, updated records and data are not processed to the exchange table.
- **Defined Filters:** Lists all the data filters supported by the class handlers.
- **Rule Editor:** The `<Filter Name>` is read only and is the name of the filter as defined by the class handler developer in the class handler method. The `<Reference Table Name>` is read only and is the technical name of the SAP database table field where the filter is applied as defined by the class handler developer. The `<Reference Field Name>` is read only and is the technical name of the SAP database table field where the filter is applied as defined by the class handler developer. The `<Data Filter Rule Key>` is an internal technical key used by the framework at runtime.
  Use the values in the *Enter Range Value* section to set the range. The `<Sign>` field is the value for the SAP range table column SIGN. The `<Option>` field is the value for the SAP range table column OPTION. The `<Low Value>` field is the value for the SAP range table column LOW. The `<High Value>` field is the value for the SAP range table column HIGH. When the `<Active Flag>` checkbox is checked, the rule is active. The *Rule List* is a table that displays a list of rules that are defined using the values in the *Rule Editor* and the *Range Value* sections.

The following screen shows that any exchange detected for the exchange object *NOTIFICATION* will be considered only if the notification is maintained in one of the roles defined in the *NOTIF_CATG* criteria.



2.2.7 Push Framework Settings

2.2.7.1 Push Scenario Definition

Push scenarios define the trigger conditions, type of data, the mobile users receiving the data, and the users for the data.

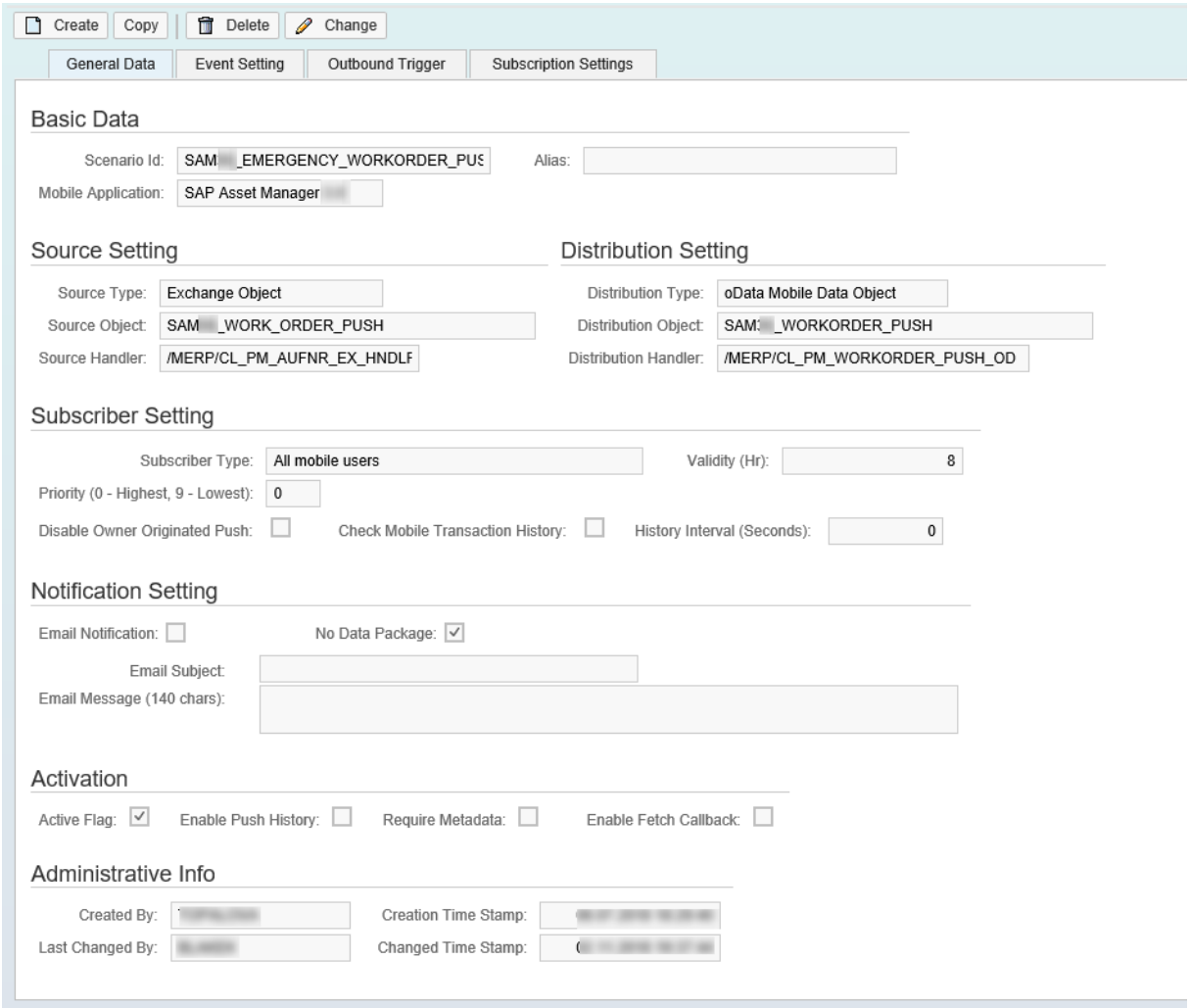
A mobile client typically synchronizes with the SAP system by initiating a synchronization request to download the latest application data from the SAP system. Some mobile applications require the SAP system to send application data or push notifications to the client when certain trigger conditions are met. If these trigger conditions are not present, the mobile client does not initiate the synchronization request.

You define trigger conditions through the creation of push scenario definitions. Use the tabs found in the *Push Scenario Definition* page to configure a push scenario. The *Push Scenario Definition* page contains the following tabs:

- General Data

- Event Setting
- Outbound Trigger
- Subscription Settings

General Data Tab



You can define the following attributes in the *General Data* tab:

- **Basic Data section:** Enter the ID of the push scenario in the required `<Scenario ID>` field, which is limited to 40 characters with namespace protection. Use either a *Y* or a *Z* namespace. Ensure that the ID is unique in the SAP system. Enter the name of the mobile application in the `<Mobile Application>` field, limited to 40 characters. Give an optional `<Alias>` to the push scenario. Multiple push scenarios can share the same alias, to allow central processing on the client side.
- **Source Setting section:** The `<Source Type>` defines how to trigger the push scenario. Two options are supported:
  - xChange Object: The push scenario is triggered when qualifying data is changed in the SAP system and change conditions defined in the xChange object are detected.

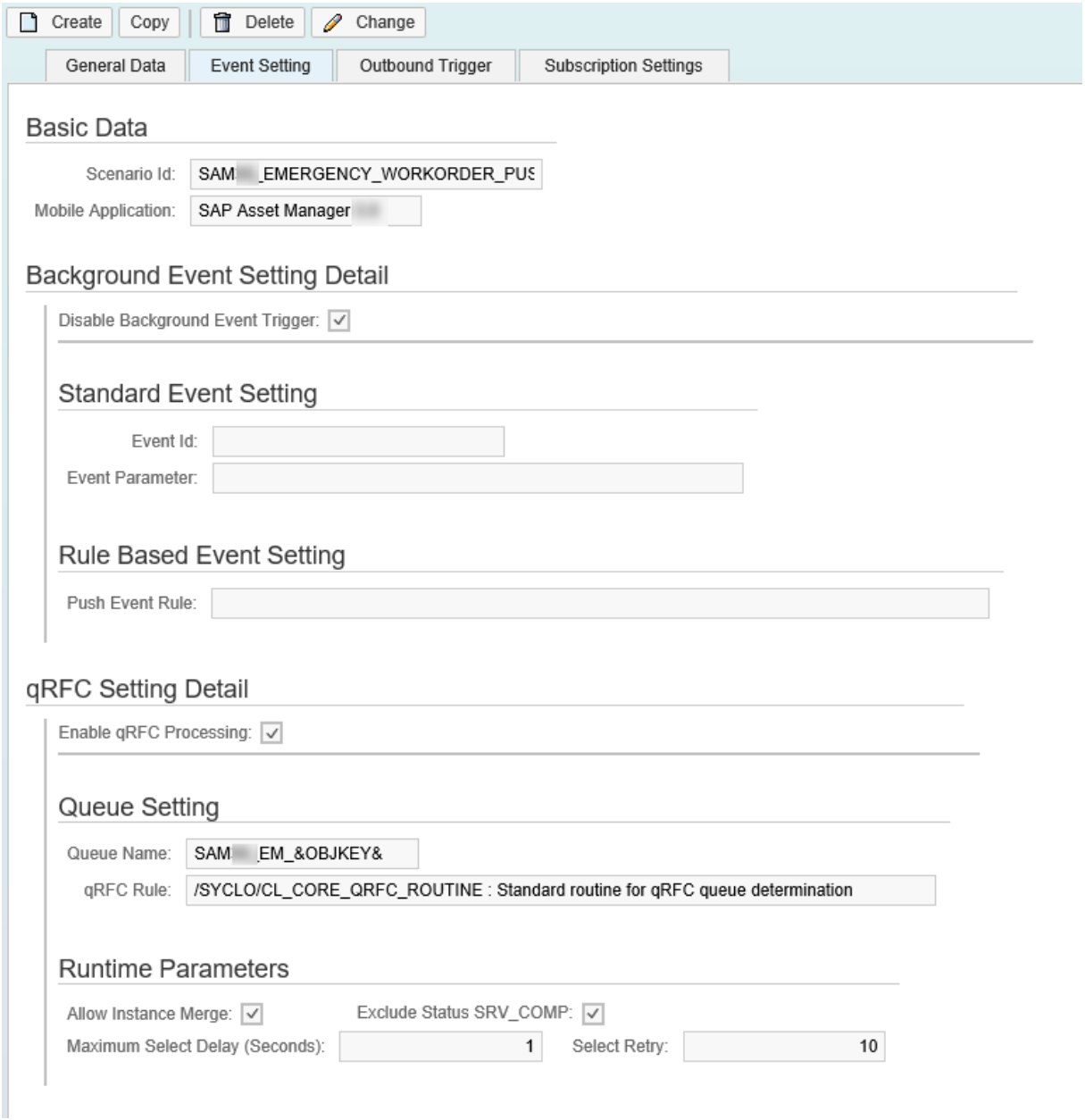
- Client on Demand Request: The push scenario is triggered based on a request from the mobile client. No data change in the SAP system is required. The client on demand request is not available for OData based mobile applications.

The `<Source Object>` applies to the source type of the xChange object. The xChange object determines the data change trigger for the push scenario. The `<Source Handler>` is the xChange handler assigned to your selected xChange object.

- **Distribution Setting section:** The `<Distribution Type>` defines what application data is sent and how recipients are determined. Two options are supported:
  - Mobile Data Object: Not supported for the SAP Service and Asset Manager application
  - OData Mobile Data Object: Use *OData Mobile Data Object* for the SAP Service and Asset Manager application. The assigned OMDO determines the recipients for the push notification.

  When you select *OData Mobile Data Object* as a Distribution Type, assign an OMDO configuration object to the OMDO in the `<Distribution Object>` field. The `<Distribution Handler>` is the OMDO handler associated with the assigned OMDO configuration object.
- **Subscriber Setting section:** The `<Subscriber Type>` determines the user pool. The user pool is used to determine who receives the push notification. Enter the amount of time the push instance generated for the push scenario is valid, in hours, in the `<Validity>` field. The `<Priority>` is the processing priority of the push instance. A push instance with a higher priority is processed first.
  When the *Disable Owner Originated Push* checkbox is checked, the user who made the change to the SAP data is excluded from the push recipient list triggered by the data change. When the *Check Mobile Transaction History* checkbox is checked, the SAP system only sends a push to the user if the user has previously downloaded the same SAP object and the object is still valid. This setting is commonly used for pushing time sensitive data such as workflow tasks to users.
  Enter the `<History Interval>`, in seconds, to determine whether the object received by the user through a previous push is still valid. No push is sent to a user who has expired push data. The *History Interval* setting is linked to the *Check Mobile Transaction History* setting.
- **Notification Setting section:** When the *Email Notification* checkbox is checked, an email notification is generated to push recipients during push processing. The email address used is based on information defined in SAP user data or mobile user leading data defined in the Administration & Monitoring Portal. If email notification is enabled, type a subject in the `<Email Subject>` field. If email notification is enabled, type your e-mail in the `<Email Message>` field.
  When the *No Data Package* checkbox is checked, no data payload is generated by push processing. Check this box to generate only push notifications. OData-based mobile applications such as SAP Service and Asset Manager only support push notifications without data payloads. Traditional, Agentry-based, mobile applications can support a data payload during push processing.
- **Activation section:** If the *Active Flag* checkbox is checked, the push is active. If the *Enable Push History* checkbox is checked, information about push recipients and data object keys are recorded in the push history table. If the *Require Metadata* checkbox is checked, technical data for the push data payload is generated. If the *Enable Fetch Callback* checkbox is checked, the mobile client retrieves data about the pushed SAP object using a regular sync request instead of precalculating the push data payload during push processing. Enabling push callback can ensure the client always receives up-to-date SAP data when receiving a push.

Event Setting Tab

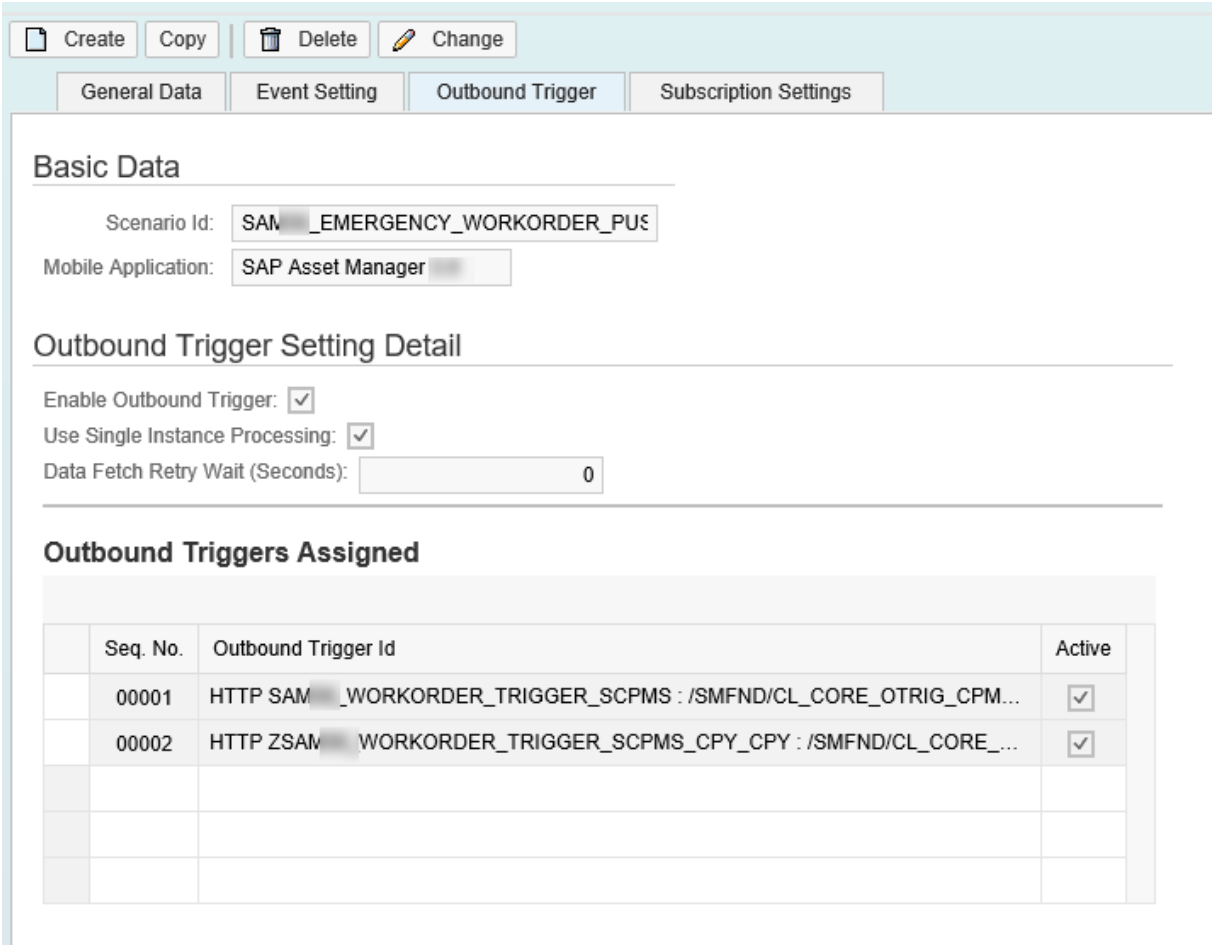


You can define the following attributes in the *Event Setting* tab:

- **Background Event Setting Detail setting:** If the *Disable Background Event Trigger* checkbox is *not* checked, a background event is raised during push processing.
- **Standard Event Setting:** The `<Event ID>` is the background event ID that is raised. The `<Event Parameter>` is the background event parameter.
- **Rule Based Event Setting:** The `<Push Event Rule>` is a routine that generates a dynamically formatted event ID and parameter based on supported runtime variables.
- **qRFC Setting Detail:** If the *Enable qRFC Processing* checkbox is checked, push processing is handled in the background as a qRFC call.

- **Queue Setting:** Enter the qRFC queue name used for push processing in the `<Queue Name>` field. The `<qRFC Rule>` is the routine that can generate a dynamically formatted qRFC queue name based on supported runtime variables.
- **Runtime Parameters:** If the *Allow Instance Merge* checkbox is checked, the creation of a new push instance cancels existing push instances for the same SAP object key that are not completed. If the *Exclude Status SRV_COMP:* checkbox is checked, push instances with a status of SRV_COMP are not reprocessed. If it is not checked, push instances with a status of SRV_COMP are reprocessed.
  Type in the time delay before processing a push sequence in the `<Maximum Select Delay>` field. The maximum select delay is used in qRFC push processing. Use this setting if there could be a time delay in the push instance registry table database update. The delay could allow the push registry data to fully update before the push processing starts.
  Type in the number of times the push processor should try to read push instance information from the push registry if the read failed into the `<Select Retry>` field.

Outbound Trigger Tab

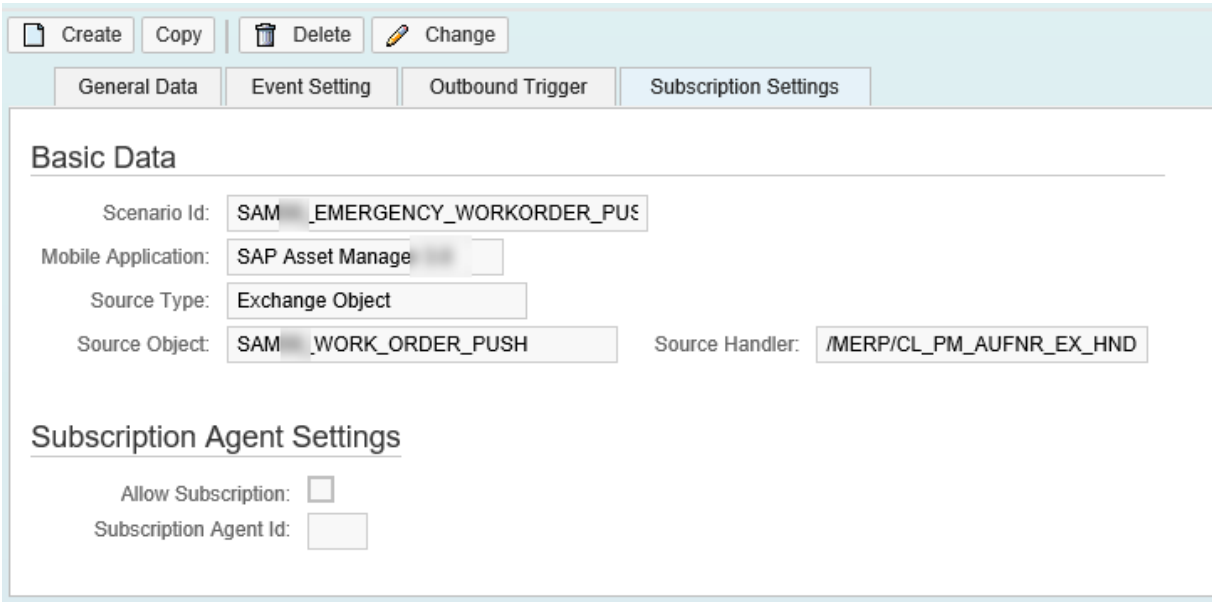


Outbound triggers handle interfacing with external systems. You can assign multiple outbound triggers to a push scenario. Assigned outbound triggers are invoked at the end of push processing, based on the assigned sequence.

Check the *Enable Outbound Trigger* checkbox to activate the selected outbound trigger.

Subscription Settings Tab

To allow an on-demand subscription based push request from the mobile client, define the subscription setting in the *Subscription Settings* tab. Subscriptions allow the mobile client to trigger a push process instead of a traditional trigger by the back end SAP system update. OData based mobile applications do not support subscription-based on-demand push configuration.



- **Allow Subscription:** Check to enable subscription-based push processing for the push scenario
- **Subscription Agent ID:** Displays the subscription agent assigned to handle the subscription request

2.2.7.2 Outbound Trigger Configuration

Outbound triggers allow a mobile application to interface with external systems such as the SAP Business Technology Platform.

You can integrate outbound triggers into one of the following mobile application processes:

- Push processing
- OData mobile data object processing

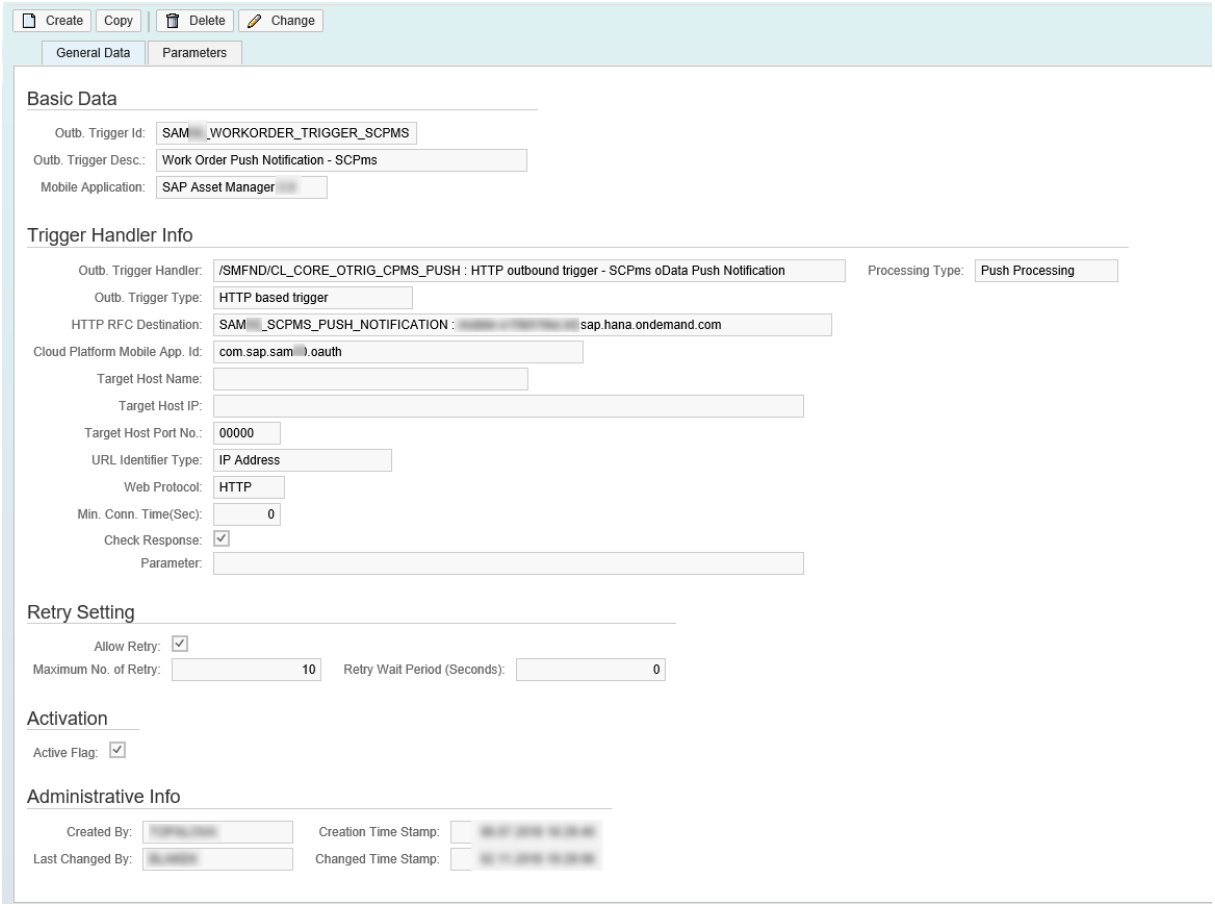
An outbound trigger can support only one of the two available processes. The process is determined by the outbound trigger handler. An outbound trigger handler can support any of the interface technologies, such as HTTP triggers, file triggers, and web service triggers.

The *Outbound Trigger* page contains the following tabs:

- General Data

- Parameters

General Data Tab



You can define the following attributes in the *General Data* tab:

Basic Data

- **Outbound Trigger ID:** Required field. Unique ID of the outbound trigger in the Y or Z namespace, limited to 40 characters.
- **Outbound Trigger Description:** Short description of the outbound trigger
- **Mobile Application:** Select your mobile application. The outbound trigger configuration detail is defined for the individual mobile application.

Trigger Handler Info

- **Outbound Trigger Handler:** ABAP OO class that provides the technical implementation for the outbound trigger. The outbound trigger handler must be a subclass of `/SYCLO/CL_CORE_XXX` You can reuse an outbound trigger handler to provide technical implementation for multiple outbound triggers.
- **Processing Type:**
- **Outbound Trigger Type:** Selection depends on the implementation of the outbound trigger handler

- **HTTP RFC Destination:** Only used when *HTTP-based trigger* is selected as the outbound trigger type. The HTTP RFC destination establishes a connection for the outbound trigger
- **Cloud Platform Mobile App ID:** used to interface with SAP Business Technology Platform Mobile Services. The ID identifies the mobile application ID that was set up in SAP BTP services.
- **Target Host Name:** Host name the outbound trigger is connecting to
- **Target Host IP:** IP address of the host to which the outbound trigger is connecting to
- **Target Host Port Number:** Port number of the host to which the outbound trigger is connecting to
- **URL Identifier Type:** Defines the information to use by the outbound trigger to make the connection
- **Web Protocol:** HTTP or HTTPS
- **Minimum Connection Time (Seconds):** Wait time before checking the response to an outbound trigger request from the remote system
- **Check Response:** If checked, a check response to an outbound trigger HTTP request is active
- **Parameter:** General purpose parameter that provides input information to the outbound trigger handler
- The following fields are only visible after other fields are selected:
  - Logical File Name: Used by outbound trigger type *File Based Trigger*
  - RFC Destination: Used by outbound trigger type *Remote Function Call*
  - RFC User ID: Used by outbound trigger type *RFC User ID*. User ID to make the RFC call.

Retry Setting

- **Allow Retry:** If checked, the outbound trigger is allowed to rerun
- **Maximum Number of Retry:** Set the maximum number of times the outbound trigger can rerun
- **Retry Wait Period (Seconds):** Set the minimum wait time between output trigger retries

Activation

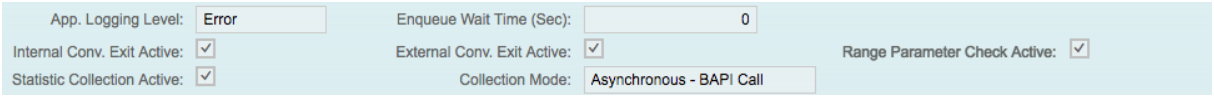
If the *Active Flag* checkbox is not checked, the outbound trigger is not enabled.

Parameters Tab

An outbound trigger handler can declare special purpose parameters. If parameters are declared, they are displayed in the *Parameters* tab. You can declare any number of parameters. A parameter can be a single field parameter or a structured record.

2.2.8 Technical Settings

Technical settings affect all components of the framework.

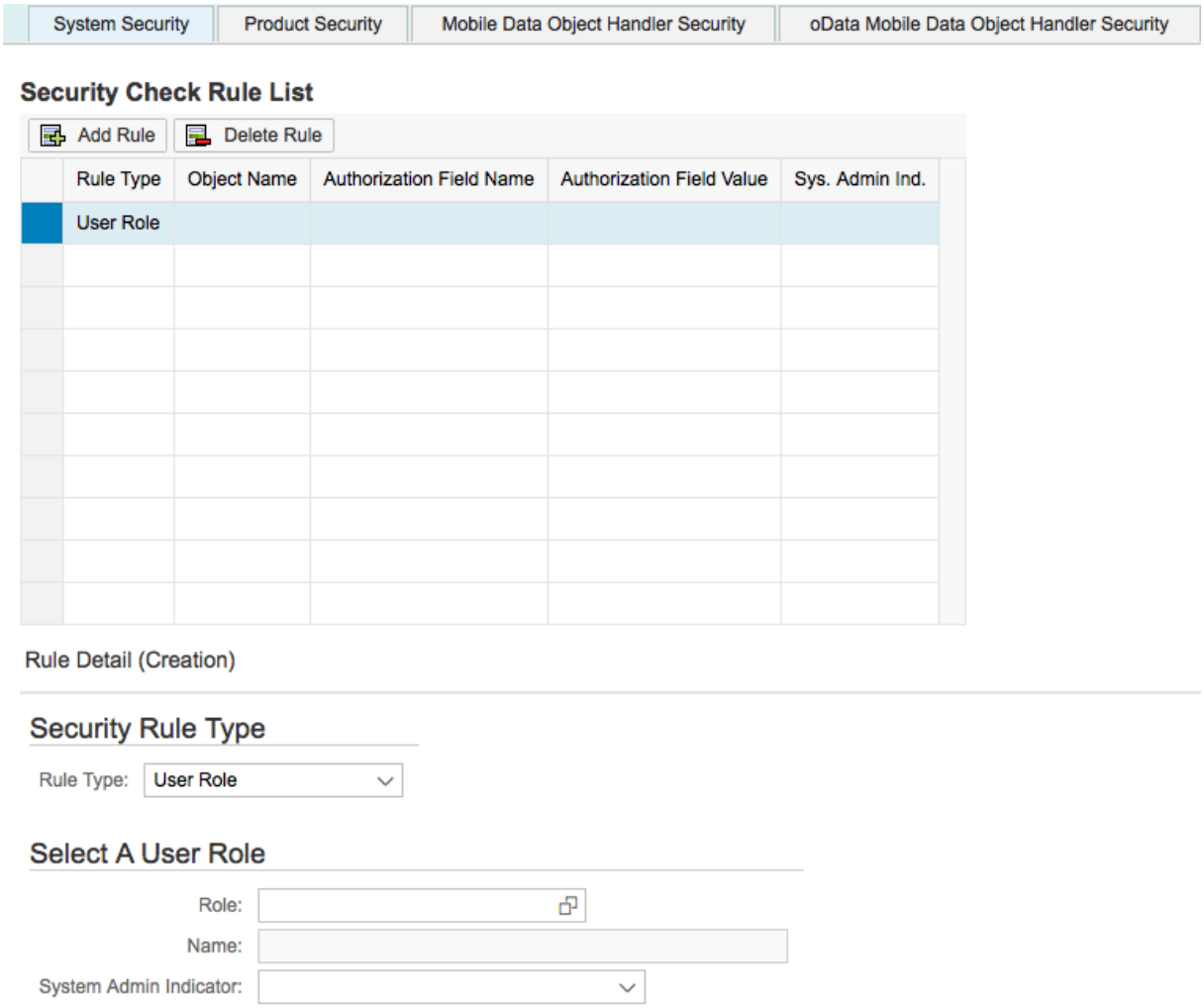


The following fields make up the Technical Settings page:

- **Application Logging Level:** Defines the logging level for all framework components. Logging entries are recorded in the SAP application log database under the object `/syclo/`. The logging levels are:
  - No logging
  - Abort
  - Error
  - Warning
  - Info
  - Debug
  - Trace
- **Enqueue Wait Time (Sec):** The Enqueue Wait Time parameter controls the number of seconds the underlying component should continue to try to access a locked SAP object in intervals of 1 second during an update by a mobile device. The update process aborts if accessing the locked object is still unsuccessful after the wait time.
- **Internal Conversion Exit Active:** When checked, the framework runtime data manager performs a standard SAP external-to-internal format conversion exit for all inbound BAPI parameters. The option is enabled by default. An application developer should only change this setting as it has a direct impact to the SAP Service and Asset Manager application.
- **External Conversion Exit Active:** When enabled, the framework runtime data manager performs standard SAP internal-to-external format conversion exit for all outbound BAPI parameters. This option is enabled by default. An application developer should only change this setting as it has a direct impact to the SAP Service and Asset Manager application.
- **Range Parameter Check Active:** When enabled, the framework runtime data manager performs checks on all SAP range parameters of inbound BAPI parameters. The SAP range parameter has the structure of SIGN, OPTION, LOW and HIGH. If SIGN and OPTION are not specified, a check routine sets SIGN to I and OTPION to EQ. This option is enabled by default. An application developer should only change this setting as it has a direct impact to the SAP Service and Asset Manager application.
- **Collection Mode:** Collection mode determines how system statistic records are written to the database. Two modes are supported currently: Synchronously and Asynchronously. When you select Synchronously, the statistics record is written to the database in real-time during BAPI calls. However, selecting this option incurs a performance penalty. Selecting Asynchronously means that statistics are collected in-memory and written asynchronously to the database at the end of the BAPI call.
- **Statistic Collection Active:** When enabled, the framework records all runtime statistics associated with the BAPI calls between the middleware server and SAP. This collection provides data for the KPI statistics collections found in the Administration portal. An application developer should only change this setting as it has a direct impact to the SAP Service and Asset Manager application.
- **Created By, Creation Time Stamp, Last Changed By, Changed Time Stamp:** The user ID and time stamps are automatically logged when a record is created or changed.

2.2.9 Mobile Authorization Settings

You can define security rule settings for the Mobile Integration Framework for SAP and mobile applications as well.



All security checks are carried out by the Mobile Integration Framework at runtime, with checks performed at the following levels:

- *System*
  Application independent. Applies to all components built on the Mobile Integration Framework.
- *Product*
  Security at the mobile application and product level
- *Mobile Data Object Handler*
  Specific to a Mobile Data Object class handler
- *OData Mobile Data Object Handler*
  Specific to an OData Mobile Data Object class handler

The following types of security rules can be defined:

- *User Role*
  Rules based on predefined user roles

- *Authorization Profile*
  Rules based on predefined authorization profiles.
- *Authorization Object*
  Rules based on predefined authorization objects.

Special Security Role with System Indicator

You can define special security rules using user roles. These security rules can be assigned with system indicators. These special security rules with system indicators are used to limit access to the ConfigPanel and Administration & Monitoring tools. The following system indicators are available:

- *System Administrator*
  If security rules are defined, only users with the required user role can have full access to the Administration & Monitoring tool.
- *System Administration – View Only*
  If security rules are defined, only users with the required user role can have read access to the Administration & Monitoring tool.
- *System Configurator*
  If security rules are defined, only users with the required user role can have full access to the ConfigPanel.
- *System Configuration – View Only*
  If security rules are defined, only users with the required user role can have read access to the ConfigPanel.

2.3 Enabling EPD Visualization

Prerequisites

1. Customize the EPD tenant. For more information, refer to the SAP Product Model View - Administration Guide guide.
2. Install and configure PMV. For more information, refer to the SAP Product Model Viewer overview.
3. Create SSAM sub-account and customize the mobile services application.
4. Create a visualization destination within the SAP Service and Asset Manager app. For more information, refer to the Configure the Mobile Application chapter of the *SAP Product Model Viewer Administration Guide*.

> **Note**
>
> For EPD visualization integration with the SAP Service and Asset Manager app to work properly, both the EPD and SAP Service and Asset Manager tenants must be set up in the same SAP BTP region.
>
> To preconfigure the SAP Product Model Viewer (PMV), refer to the Overview chapter of the *SAP Product Model Viewer Administration Guide*.

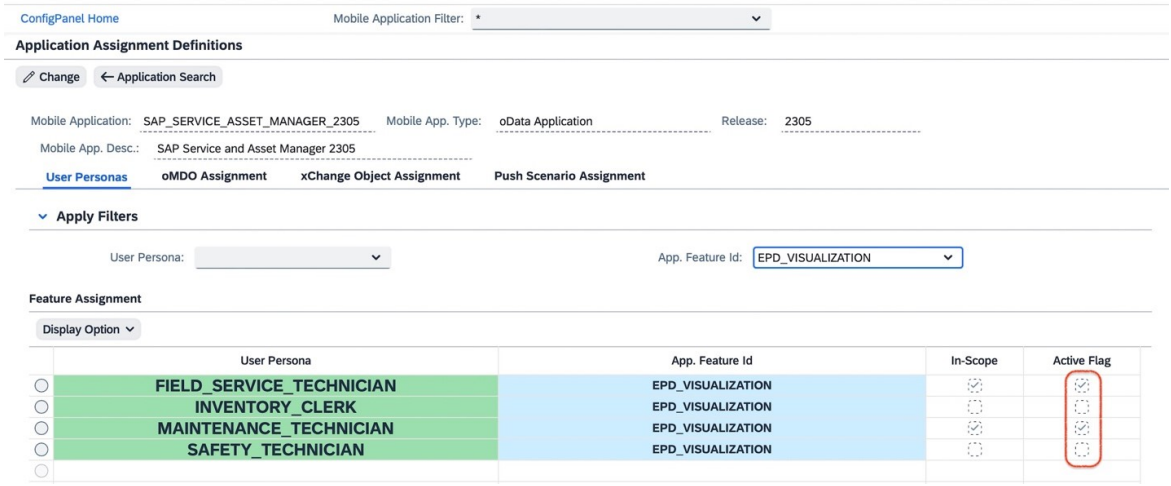
Context

EPD visualization in the SAP Service and Asset Manager is available by enabling the feature in the configuration panel. The feature is only available for the Maintenance Technician and Field Service Technician personas.

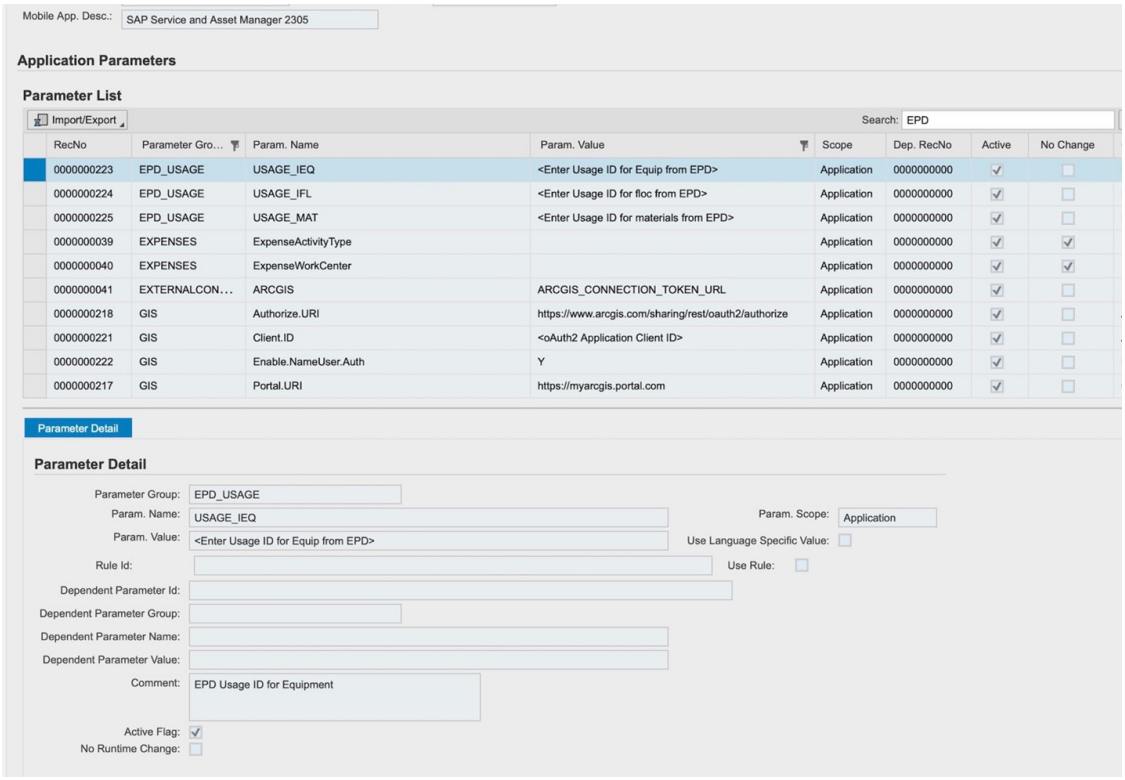
Procedure

1. To enable or disable the feature, check or uncheck the *Active Flag* box and save.

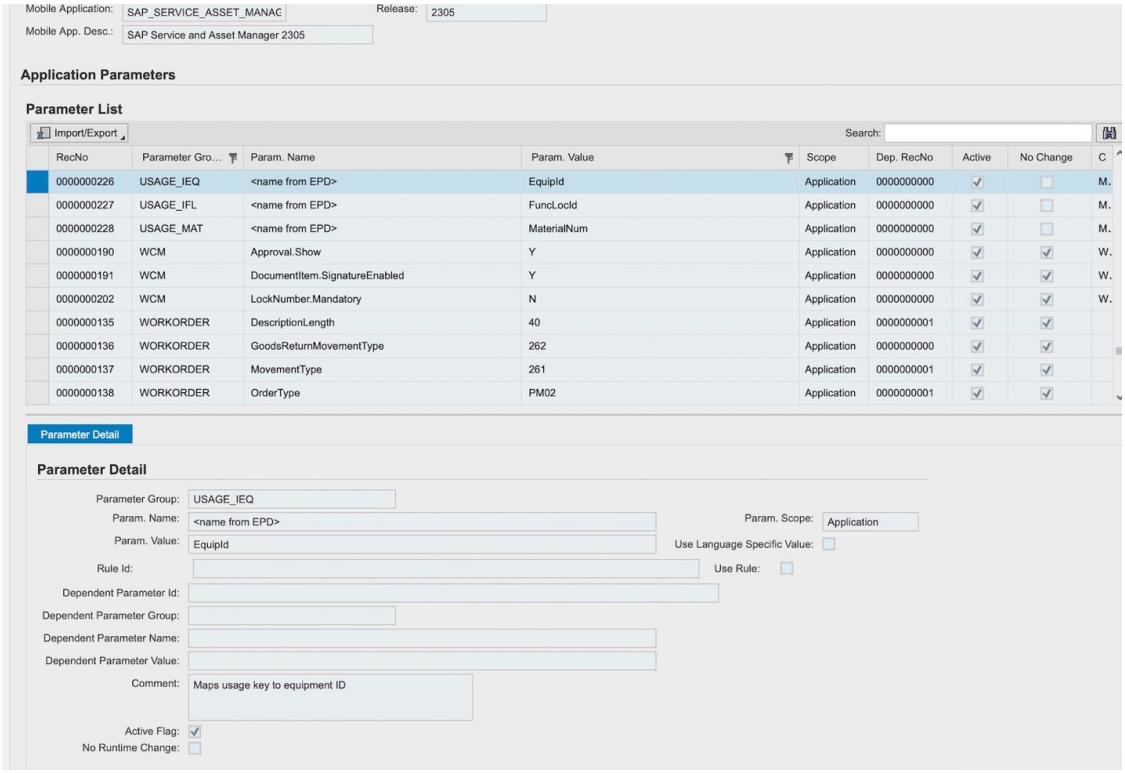
   > **Note**
   >
   > By default, this feature is enabled in *Configuration Panel* > *Component Assignment* > *<Select Mobile Application>* > *User Personas tab*.



2. To use EPD visualization in the SAP Service and Asset Manager app,maintain the application parameters in the ConfigPanel. Navigate to *Application Configuration* > *<select Mobile Application>* > *Parameters tab*.
3. Depending on which object is configured in the EPD, choose an option:
   - **To maintain usage ID:** *Param. Value:* Usage ID for equipment/functional location/material from EPD (example values: *As-Maintained*/*As-Installed*/*As-Designed*)



- To maintain key usage for objects:
  - **Param. Name:** Usage key from the EPD customization (ex: *Equipment_ID*/*FuncLocID*/*Material*)
  - **Param. Value:** Key field of the object or entity type. Note that *EquipID*, *FuncLocId*, and *MaterialNum* are configured by default.



2.4 Low Code No Code Support

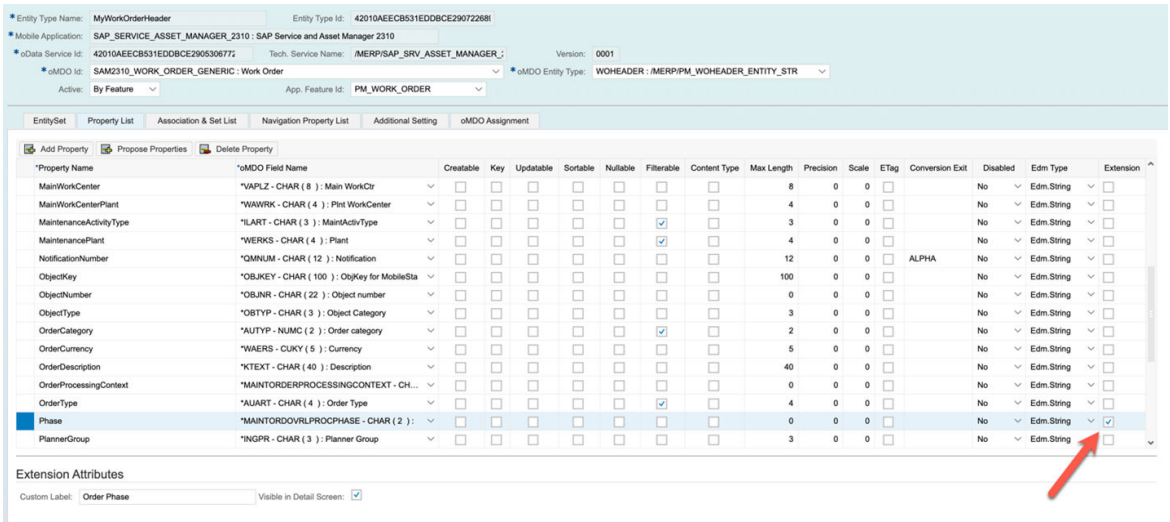
Context

The object detail screens in SAP Service and Asset Manager can be modified to include properties of the oData entity set that are not part of the detail screen out of the box. To do this, simply change the backend configuration and then synchronize with the mobile client.

Procedure

1. Log on to the SAP system and execute the transaction code /SYCLO/CONFIGPANEL.
2. Log on to the Configuration Panel.
3. Navigate to *oData Model Configuration*.
4. Open *SAP_SERVICE_ASSET_MANAGER_2310* and find the entity you want to change on the mobile client.
5. Navigate to the *Property List* tab and select the new oData entity set property that should appear on the object detail screen. (Custom Z properties are also supported).
6. Click the *Extension* check box.

   The *Extension Attributes* section appears.



7. Enter the label to be used on the mobile client for this property if necessary. If this value is empty, the default data item field label associated with the oMDO entity type structure component will be used.
8. Click the *Visible in Detail Screen* check box.
9. Save changes and select a customization request if necessary.
10. Synchronize from a mobile device.

11. The property will now appear on the detail screen of the corresponding object.

2.5 Optimized Delta Synchronization

The SAP Service and Asset Manager application performs two types of synchronization: initial synchronization and delta synchronization. Initial synchronization is performed either when the user logs in to the application for the first time or after the application is reset. Any subsequent synchronization performed after a successful initial synchronization and before the app is reset is a delta synchronization, which typically uploads any transactions created locally on the mobile device to the backend system and downloads delta from the backend to the mobile device for entities that have changed since the last synchronization.

As of SAP Service and Asset Manager version 2310, the application is activated for optimized delta synchronization by default. In this case, when the user initiates delta synchronization, an attempt is made to upload transactions, if any. An interactive call then identifies entities that need to be updated or have changed since the last synchronization. Only for these changed entities will delta synchronization be started, thus optimizing the duration of delta synchronization.

Qualifying entities for delta synchronization

Since the last synchronization completed, entities are qualified for delta synchronization as follows:

1. An entity that has a new dependent queue
2. An entity that has a triggered exchange
3. An entity whose configuration includes delta key calculation on every delta synchronization
4. An entity for which periodic update is enabled and expired
5. Entities that are marked as dependent for the above qualified entities from 1, 2, 3, and 4
6. Entities from 5 that are active by feature or persona for the respective user

Optimized delta synchronization will be equivalent to normal delta synchronization if:

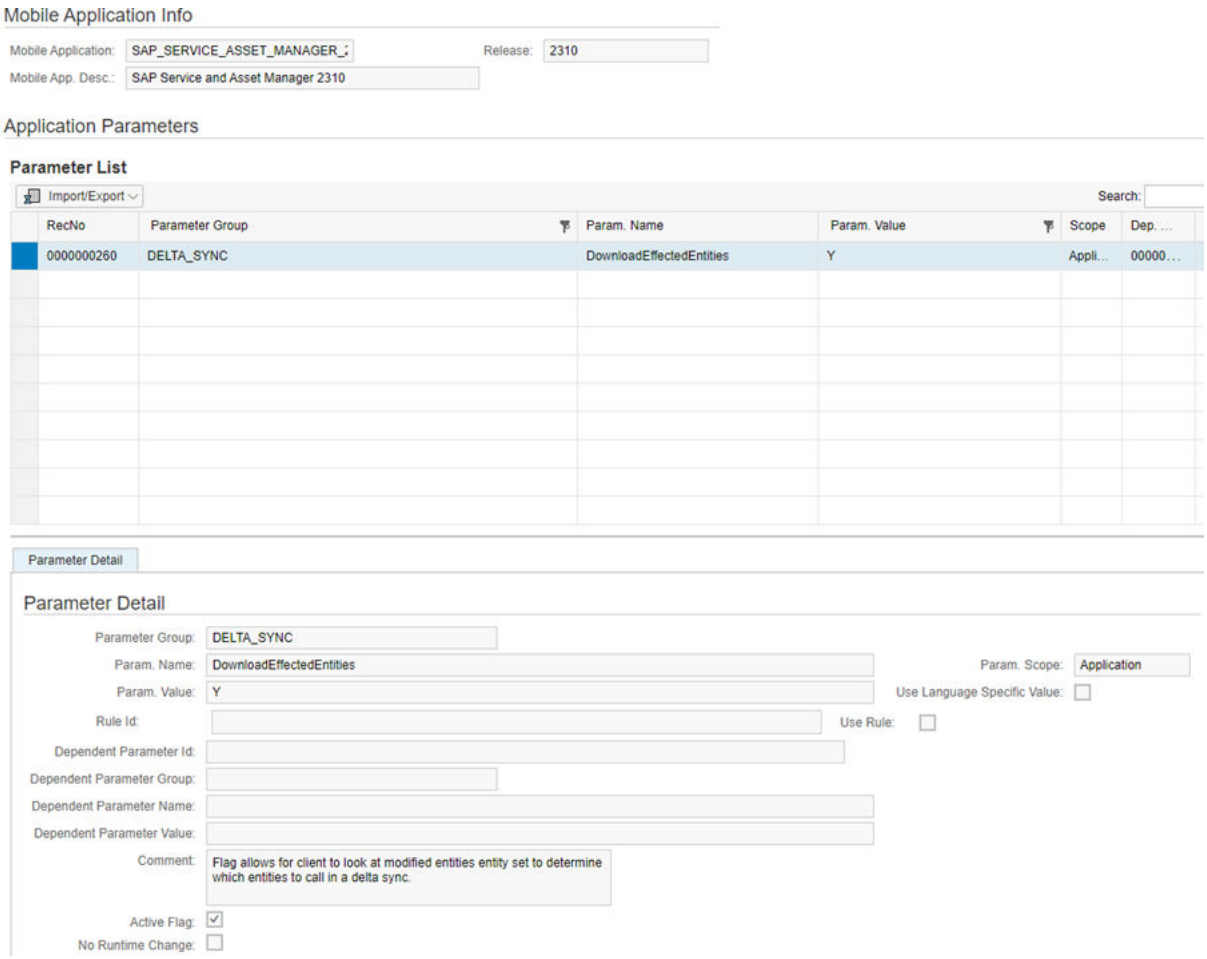
1. The application cannot detect the completion time of the user's last synchronization
2. The application's *DownloadEffectedEntities* parameter is set to N (off)
3. The application initiated synchronization due to a user change, for example, user A logged out and user B logged in, which initiated synchronization

The mobile configuration parameter must be set as follows (available in SAP Service and Asset Manager 2310 version):

Parameter group: DELTA_SYNC

Parameter name: DownloadEffectedEntities

Param. Value: Y

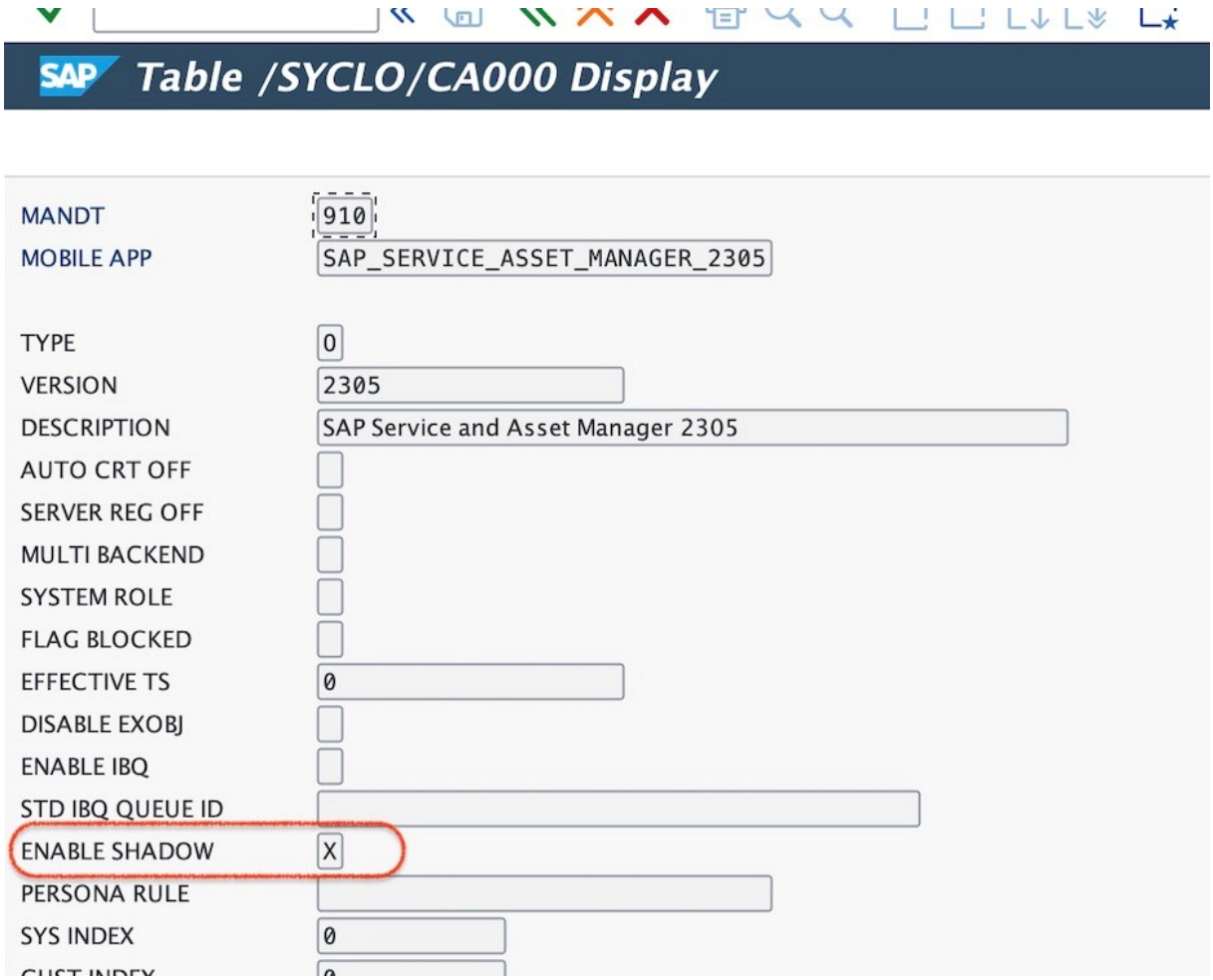


> **Note**
>
> To disable optimized delta synchronization, set the *DownloadEffectedEntities* parameter value to N and save it.

2.6 Shadow Table

As of the release of SSAM 2305, support for shadow table is now available. Any custom changes will be stored in separate sets of tables, unlike standard configuration tables that use the MAIF framework. Shadow table support is only available for oData applications. Any custom application created using the configuration panel will be marked as supporting shadow tables.

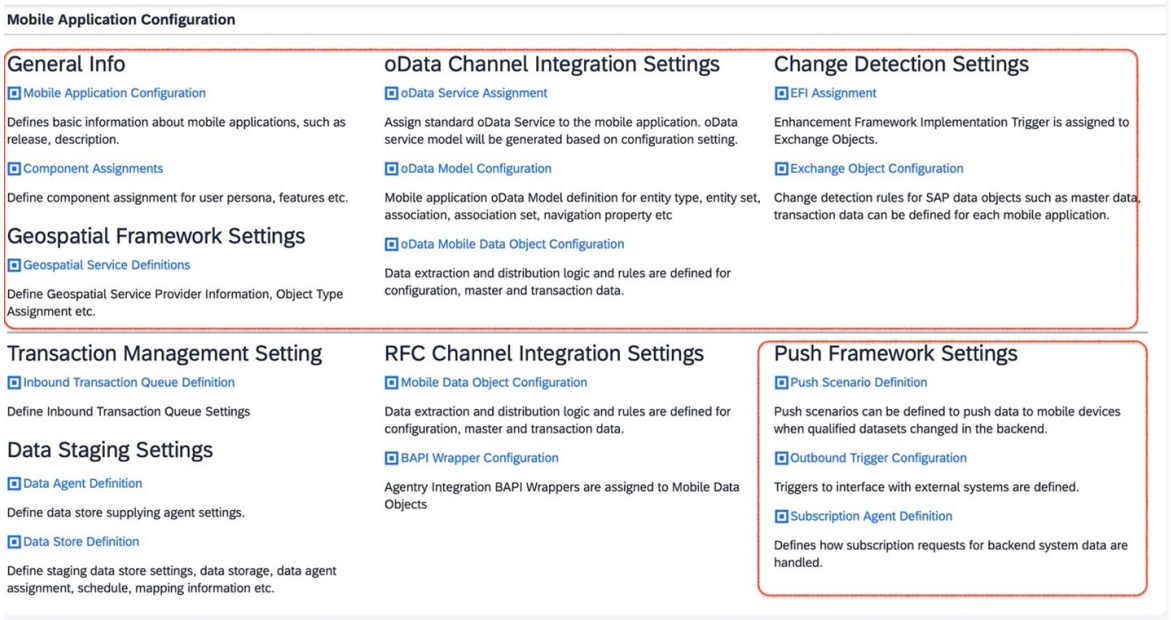
The *Enable Shadow* checkbox in the mobile application tables (/SYCLO/CA000 and /SYCLO/YCA000) is introduced to determine whether or not the application has enabled shadow table support:



If the application is enabled in shadow table mode, all user configuration changes will be saved in the shadow tables.

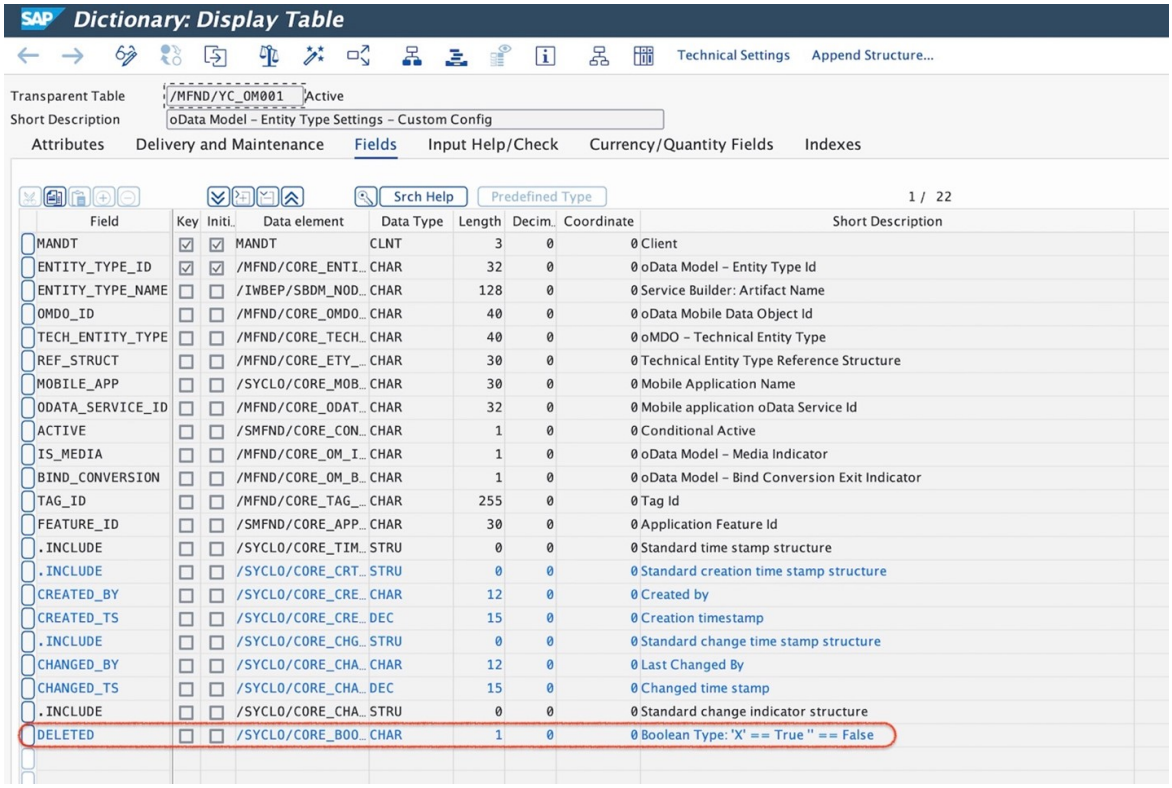
The following configuration fields from the configuration panel can be supported by shadow tables:

- Mobile Application Configuration
- Component Assignments
- Geospatial Service Definitions
- oData Service Assignment
- oData Model Configuration
- oData Mobile Data Object Configuration
- EFI Assignment
- Exchange Object Configuration
- Push Scenario Definition
- Outbound Trigger Configuration
- Subscription Agent Definition



A shadow table has the same name as a standard configuration table, with an extra *Y* in the table name. For example, the standard configuration table for an entity type is /MFND/C_OM001. The corresponding shadow table is /MFND/YC_OM001.

The shadow table has all the fields as in the standard configuration table, with an extra *DELETED* field used to indicate whether or not the standard configuration has been deleted. The configuration panel procedures combine the standard and custom configurations and return a consolidated result for display.



Field	Key	Initi.	Data element	Data Type	Length	Decim.	Coordinate	Short Description
MANDT	☑	☑	MANDT	CLNT	3	0	0	Client
ENTITY_TYPE_ID	☑	☑	/MFND/CORE_ENTI...	CHAR	32	0	0	oData Model – Entity Type Id
ENTITY_TYPE_NAME	☐	☐	/IWBEP/SBDM_NOD...	CHAR	128	0	0	Service Builder: Artifact Name
OMDO_ID	☐	☐	/MFND/CORE_OMDO...	CHAR	40	0	0	oData Mobile Data Object Id
TECH_ENTITY_TYPE	☐	☐	/MFND/CORE_TECH...	CHAR	40	0	0	oMDO – Technical Entity Type
REF_STRUCT	☐	☐	/MFND/CORE_ETY_...	CHAR	30	0	0	Technical Entity Type Reference Structure
MOBILE_APP	☐	☐	/SYCLO/CORE_MOB...	CHAR	30	0	0	Mobile Application Name
ODATA_SERVICE_ID	☐	☐	/MFND/CORE_ODAT...	CHAR	32	0	0	Mobile application oData Service Id
ACTIVE	☐	☐	/SMFND/CORE_CON...	CHAR	1	0	0	Conditional Active
IS_MEDIA	☐	☐	/MFND/CORE_OM_I...	CHAR	1	0	0	oData Model – Media Indicator
BIND_CONVERSION	☐	☐	/MFND/CORE_OM_B...	CHAR	1	0	0	oData Model – Bind Conversion Exit Indicator
TAG_ID	☐	☐	/MFND/CORE_TAG_...	CHAR	255	0	0	Tag Id
FEATURE_ID	☐	☐	/SMFND/CORE_APP...	CHAR	30	0	0	Application Feature Id
.INCLUDE	☐	☐	/SYCLO/CORE_TIM...	STRU	0	0	0	Standard time stamp structure
.INCLUDE	☐	☐	/SYCLO/CORE_CRT...	STRU	0	0	0	Standard creation time stamp structure
CREATED_BY	☐	☐	/SYCLO/CORE_CRE...	CHAR	12	0	0	Created by
CREATED_TS	☐	☐	/SYCLO/CORE_CRE...	DEC	15	0	0	Creation timestamp
.INCLUDE	☐	☐	/SYCLO/CORE_CHG...	STRU	0	0	0	Standard change time stamp structure
CHANGED_BY	☐	☐	/SYCLO/CORE_CHA...	CHAR	12	0	0	Last Changed By
CHANGED_TS	☐	☐	/SYCLO/CORE_CHA...	DEC	15	0	0	Changed time stamp
.INCLUDE	☐	☐	/SYCLO/CORE_CHA...	STRU	0	0	0	Standard change indicator structure
DELETED	☐	☐	/SYCLO/CORE_BOO...	CHAR	1	0	0	Boolean Type: 'X' == True '' == False

With shadow table support, customers can upgrade SSAM versions without affecting previously created custom configurations. The migration tool has been enhanced to support migration of shadow table data.

2.7 Removal of Dependent Objects Based on Lead Objects

SAP Service and Asset Manager downloads data to mobile by using distributon mode. One of the supported distributon modes is Dependency mode. When Dependency mode is activated for oMDO, objects get qualified by the lead object. With this new feature introduced, when the last lead object is removed from device, the associated dependent object will also be removed.

Feature Description

See the following example and explanation for this feature:

Example: *Workorder 450000001* references *technical object 100000010* and the required dependency configurations are active in the mobile applicaton configuration.

- When the *workorder 450000001* qualifies for the mobile device, the workorder oMDO generates dependency queue to equipment oMDO. When the equipment oMDO performs key calculation, it qualifies the *technical object 100000010* for download.
- With this feature, when the life cycle of *workorder 450000001* is completed from the mobile device and the object gets removed from it, the *technical object 100000010* (which was downloaded as dependent object to *workorder 450000001*) will also be removed.

> ⓘ Note
>
> If the same dependent object is referenced by more than one lead objects (e.g., workorder and notification), the dependent object will be removed **only when the last lead object life cycle is completed and removed from mobile device**.

Supported Objects

Removal of dependent objects based on lead object removal is supported **only** for the following objects using standard handlers:

PM Objects in SAP Service and Asset Manager	Handlers
Notification	`/MERP/CL_PM_NOTIFICATION_OD`
Functional Location	`/MERP/CL_PM_FUNCLOC_OD`
Equipment	`/MERP/CL_PM_EQUIPMENT_OD`
Measuring Point	`/MERP/CL_PM_MEASURING_POINT_OD`
Measurement Document	`/MERP/CL_PM_MEASUREMENT_DOC_OD`

Standard Support for SAP Service and Asset Manager 2310

From SAP Service and Asset Manager 2310, all configurations required for this feature are available as standard, with the following exception:

- If the LITE version of oMDO is used for Equipment and/or Functional Location, the removal of dependent feature is deactivated. To switch it manually, refer to Support for LITE Version of Technical Objects [page 79].

If you want to use this feature on SAP Service and Asset Manager versions released **before 2310**, please consider the following:

- In addition to the existing standard configuration (and other notes it there are any), you must apply the following manual configuration changes:
  1. For configuration steps, see Section 3 (*High-Level Steps to Enable Support for SAP Service and Asset Manager 2305 or lower*) of the attached document in 3378889 SAP note.
  2. After the configuration changes are done, reset the application.

Design and Limitations

Please consider the following information and limitations regarding the feature:

- Any changes to the configuration and/or offline files require a reset and initial sync of the application.
- If entries in the user dependent list are modified or deleted manually, or other than the standard oMDO is used, the feature may not work as intended.
- This feature keeps track of objects that are downloaded **only via dependency queue**. If the same object has qualified via another distribution mode, this feature will not work.
- Changes of referenced dependent objects after the lead object was downloaded to the mobile device are treated as a new dependent object for the lead object.

  > ⓘ Note
  >
  > For example, the lead object references to dependent object "A" and it has already been downloaded to the mobile device. Then in the back end if the referenced dependent object is changed from "A" to "B", the delta sync will refresh the lead object with the newly referenced dependent object "B". However, the removal of dependent object "A" happens only on reset of mobile application. The removal of dependent object "B" happens when the lead object is removed from device.

- If the implementation involves client state keys and/or origin dependent keys format change (either through code and/or configuration), this feature may not work.
- Unless explicitly stated, no other objects support the dependency removal without a reset of the mobile application.
- With 2310, LITE version of oMDO is introduced for Equipment and Functional Location objects. While the regular oMDO for Equipment and Functional Location downloads the full-blown data for each qualified object that is associated with workorder or notification, the LITE version downloads only the minimum details of technical object (i.e., without any dependencies like documents, classification, characteristics, measuring points, measurement documents etc.).
  - As a result, when the LITE version of oMDO is active for Equipment and/or Functional Location as standard configuration for the mobile application or enabled via manual configuration, the removal of dependent object for Equipment and Functional Location will be deactivated.
    - If this feature was manually enabled for lower versions of SAP Service and Asset Manager and you want to deactivate it, refer to the 3368555 SAP note.

- If you want to activate the removal of dependent object for Equipment and/or Functional Location, because the LITE version is not used in implementation, refer to Support for LITE Version of Technical Objects [page 79] to activate it.

2.8 Support for LITE Version of Technical Objects

From SAP Service and Asset Manager 2310, LITE version of technical objects is activated as standard configuration (i.e., LITE version of oMDO exists as standard for Equipment and Functional Location objects). As a result, technical objects associated with work orders or notifications are downloaded with full-blown data and others have only minimum data (LITE version). When new jobs qualifiy with LITE version of technical object, the application automatically downloads the full-blown data for the respective technical object along with the workorder or notification download.

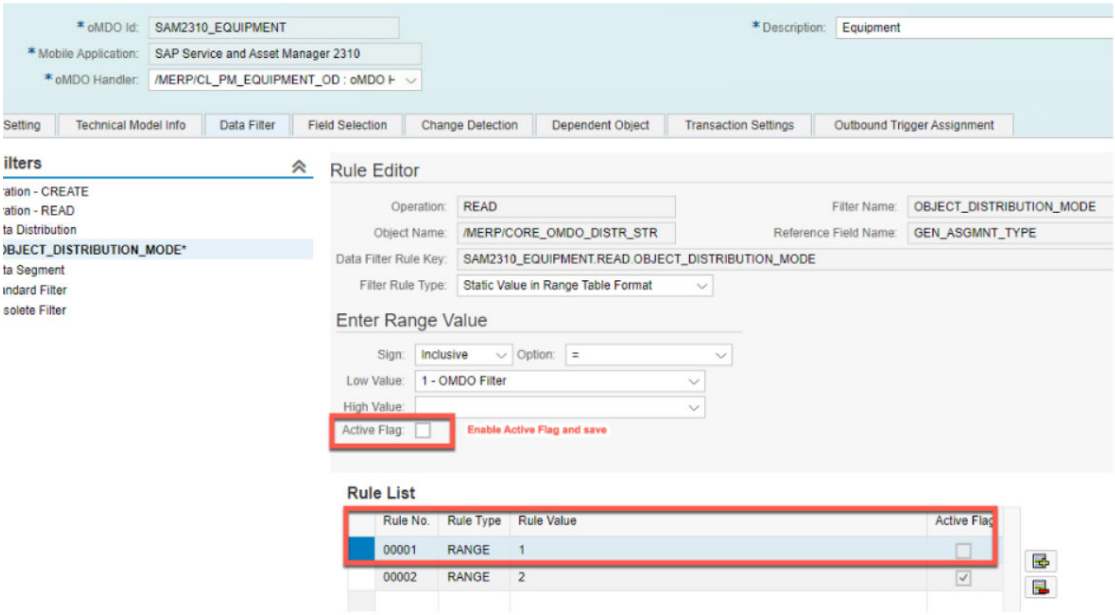
Context

If the LITE version of technical objects is not preferred for the implementation (i.e., implementation prefers full-blown data for all technical objects downloaded to the device), perform the following steps to switch:

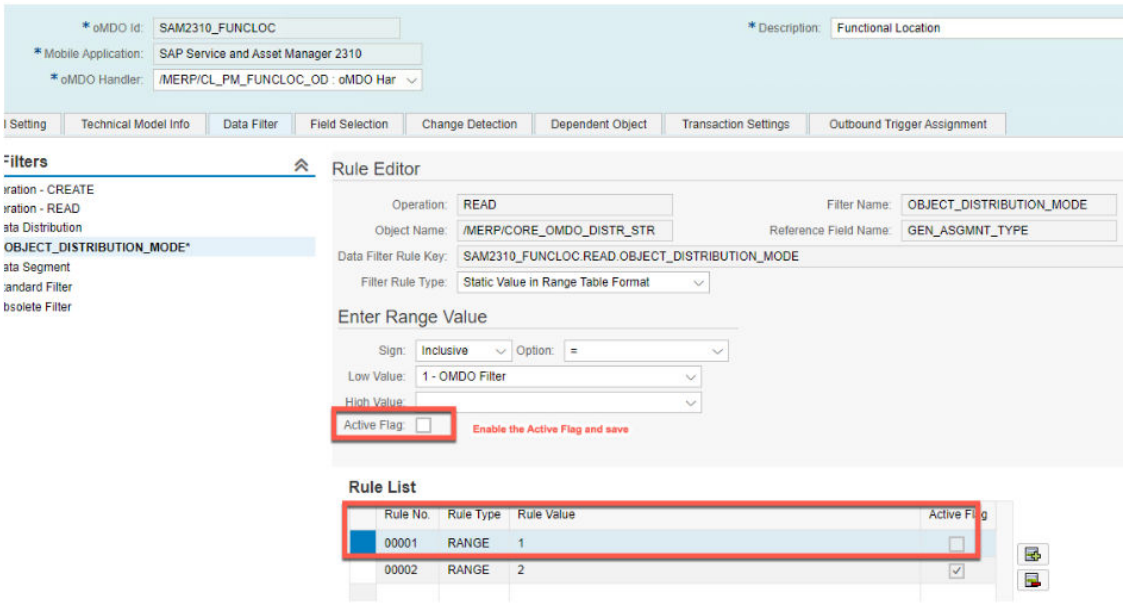
> **Note**
>
> Switching from LITE version to full-blown version is relevant only for 2310 version.

Procedure

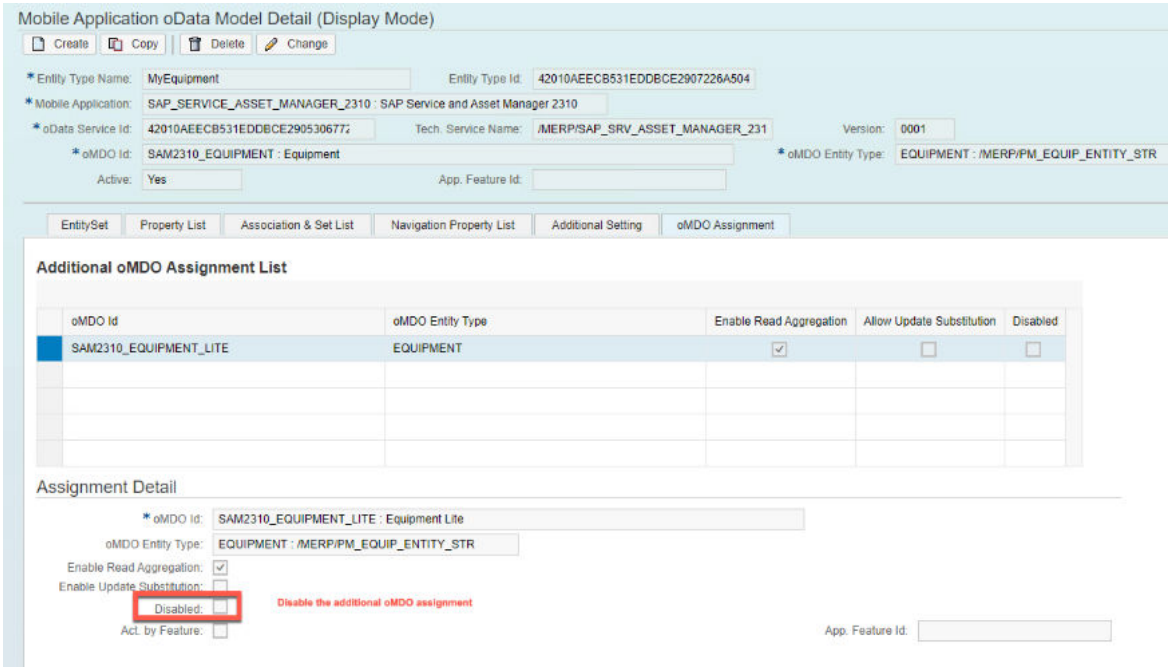
1. Disable additional oMDO assignment settings for the following oModels:
   - MyEquipment
   - MyEquipLongText
   - MyFunctionalLocation
   - MyFuncLocLongText
2. Enable oMDO data distribution filters for the following oMDOs:
   - *SAM2310_EQUIPMENT*



- *SAM2310_FUNCLOC*



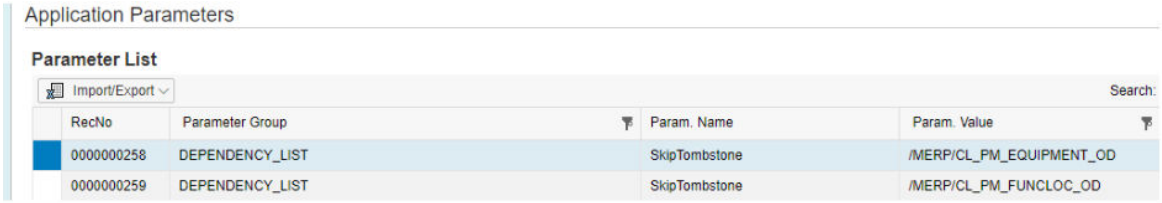
3. Disable the additional oMDO assignment for the *MyEquipment* entity type and click *Save*.



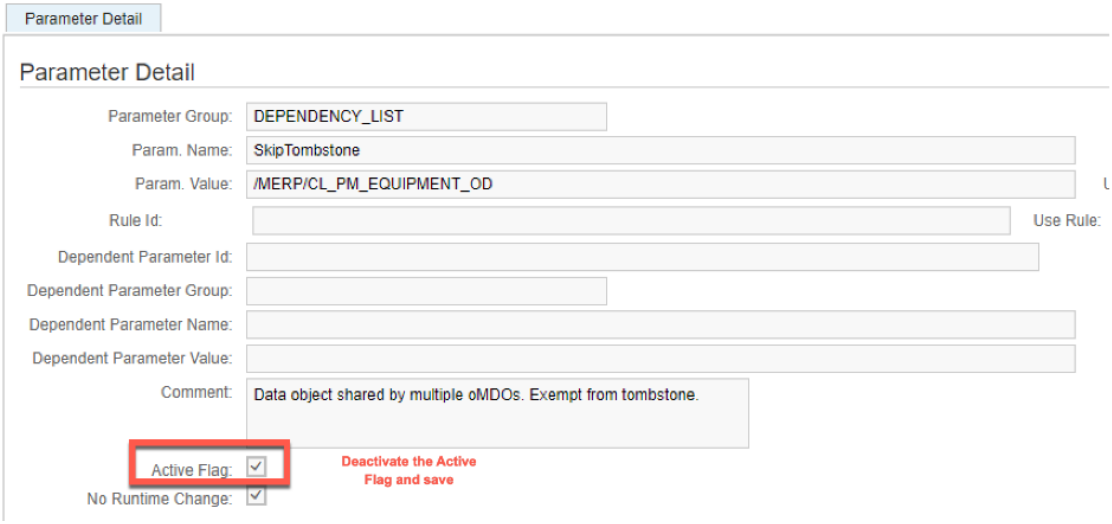
4. Repeat the previous step for all the entity types mentioned in Step 1 [page 79].

> ### Note
>
> If a similar configuration is enabled for other application versions, perform this step for those entity types as well.

5. In the respective mobile application parameter configuration, deactivate the following parameters:



- Deactivate the following parameter by unticking the *Active Flag* checkbox and save the changes. Repeat this step for all parameter values shown above.



6. When all changes are done, reset the mobile application and perform an initial sync.
7. Before productive migration, verify that all required data is downloaded and relevant functonalities work as intended.

3 Mobile Add-On Configuration Panel Common Procedures

3.1 Configuring Personas and Features Overview

Persona and feature assignments determine the data downloaded to the mobile client application.

The entity types below are introduced in SAP Service and Asset Manager to return persona- and feature-related configurations to the mobile client. The client logic looks at these entity types to check user personas assignment or assignments and to see if features are enabled. This is used in the mobile client application to determine which user interface elements should appear on certain screens and sections.

- **UserFeature:** In the /MERP/SAP_ONLINE_LOOKUP_EXT_XXXX service, this entity returns a list of features that are applicable based on user personas. With the support of UserFeature, enable parameters are no longer defined in AppParameters.
- **UserPersonas:** In the MERP/SAP_ONLINE_LOOKUP_EXT_XXXX service and /MERP/SAP_SRV_ASSET_MANAGER_XXXX service, this entity returns user persona assignment.
- **UserSyncGroupDetails:** In the /MERP/SAP_ONLINE_LOOKUP_EXT_XXXX service, this entity returns list of entity sets that are applicable based on user personas.

Previously, the application would download all data regardless of enabled or disabled features. This isn't optimal and can impact the performance of initial and delta syncs. Now, during the initial sync, the mobile client makes an online call to fetch the logged on user's persona assignment and entities needed to request in subsequent calls. The only data downloaded is data needed for the persona:

Stat...	Service Call Info	Method	Proc. Time	Appl. Time	Non-GW T...	Req. Size	Resp. Size	Form
■	/MERP/SAP_ASSET_MANAGER_2110/$batch	POST	4,739	2,040	0	4,910	37,213,4...	mixe
■	/MERP/SAP_ASSET_MANAGER_2110/$batch	POST	795	390	0	1,629	5,009,567	mixe
■	/MERP/SAP_ASSET_MANAGER_2110/$batch	POST	4,659	2,471	0	6,469	27,056,0...	mixe
■	/MERP/SAP_ASSET_MANAGER_2110/$batch	POST	1,980	961	0	16,681	4,332,311	mixe
■	/MERP/SAP_ASSET_MANAGER_2110/OpenSyncSession?SAPProd...	POST	66	34	0	0	520	json
■	/MERP/SAP_ASSET_MANAGER_2110/$metadata?sap-language=...	GET	290	2	0	0	1,045,062	xml
■	/MERP/SAP_ONLINE_LOOKUP_EXT_2110/UserSyncGroupDetails	GET	80	38	0	0	246,904	xml
■	/MERP/SAP_ONLINE_LOOKUP_EXT_2110/UserPersonas	GET	61	40	0	0	1,517	xml
■	/MERP/SAP_ONLINE_LOOKUP_EXT_2110/$metadata	GET	14	2	0	0	0	
■	/MERP/SAP_ONLINE_LOOKUP_EXT_2110/	GET	16	2	0	0	5,766	xml
■	/MERP/SAP_ONLINE_LOOKUP_EXT_2110/$metadata	GET	27	1	0	0	48,669	xml
■	/MERP/SAP_ONLINE_LOOKUP_EXT_2110/	GET	27	2	0	0	5,766	xml
■	/MERP/SAP_ONLINE_LOOKUP_EXT_2110/$metadata	GET	30	2	0	0	48,669	xml
■	/MERP/SAP_ONLINE_LOOKUP_EXT_2110/	GET	18	2	0	0	5,766	xml

Use the following topics to configure your personas and features in your mobile application:

- Configuring Personas [page 87]
- Configuring Features [page 90]
- Configuring Component Assignments [page 97]

3.1.1 Supported Features for SAP Service and Asset Manager for Windows

SAP Service and Asset Manager for Windows allows a single user to be assigned the Maintenance Technician Persona role. The table below shows the main functions of the persona.

> **Note**
>
> For more information concerning features supported when using SAP Service and Asset Manager for Windows, refer to the note https://launchpad.support.sap.com/#/notes/3268538.

Feature	Feature Description	Maintenance Tech Persona	
		Applicable	Default Enabled
CA_ATTACHMENT	Attachment support for business objects (DMS / BDS / GOS)	✓	Y
CA_BUSINESS_PARTNER	Business partner	✓	N
CA_CLASSIFICATION	Classification and characteristics for technical objects	✓	Y
CA_CORE_DATA	Core data including org structure (ex: Plant, Work Center)	✓	Y
CA_CREATE_TECH_OBJECT	Create equipment and functional location	✓	Y
CA_NO_HISTORY	Notification history	✓	Y
CA_TECH_OBJECT	Equipment and functional location master data	✓	Y
CA_WO_HISTORY	Work order history	✓	Y
CS_NOTIFICATION	Service notifications	⊘	

Feature	Feature Description	Maintenance Tech Persona	
		Applicable	Default Enabled
CS_SERVICE_ORDER	Service orders	⊘	
HR_TIMESHEET	CATS timesheet	✓	N
IAM_INDICATORS	PdMS indicators	✓	N
PM_CLOCK_IN_CLOCK_OUT	Clock In / Clock Out	✓	N
PM_CONFIRMATION	PM confirmations for time recording	✓	Y
PM_LINEAR_ASSET_MANAGEMENT	Linear Asset Management (LAM)	✓	N
PM_MEASUREMENT	Measurement readings	✓	Y
PM_NOTIFICATION	PM notifications	✓	Y
PM_OBJECT_LIST	Object lists on work orders	✓	Y
PM_PRT	Production, Resources, and Tools (PRT)	✓	Y
PM_SUPERVISOR_MODE	Supervisor mode	✓	Y
PM_WORK_ORDER	PM work orders	✓	Y

3.1.2 Usage Types

Each persona is categorized by type of usage, either *Professional* or *Standard*. This determines the Full Use Equivalent (FUE) valuation according to the updated licensing model for SAP Service and Asset Manager.

Professional type of usage is one (1) FUE and Standard is one-half (1/2) FUE. In the base configuration, Maintenance Technician is defined as a professional user, and Field Service Technician and Inventory Clerk are

defined as standard users. Customers can also define themselves as custom personas, which is considered a professional usage type.

> **Note**
>
> The usage type does not necessarily depend on the persona, but rather on the available feature sets. For example, a single FUE is the maximum value for any individual user that allows access to all available features, which means that the professional usage type is configured with more feature support than the standard usage type.
>
> The key difference is when a customer uses a standard Mobile Application Integration Framework (MAIF) app, such as SAP Service and Asset Manager, and only needs inventory or field service functionality. In this case, one (1) FUE could cover two users since those personas are standard with only one-half (1/2) FUE value.

The table below shows what type of usage can be configured for different types of application.

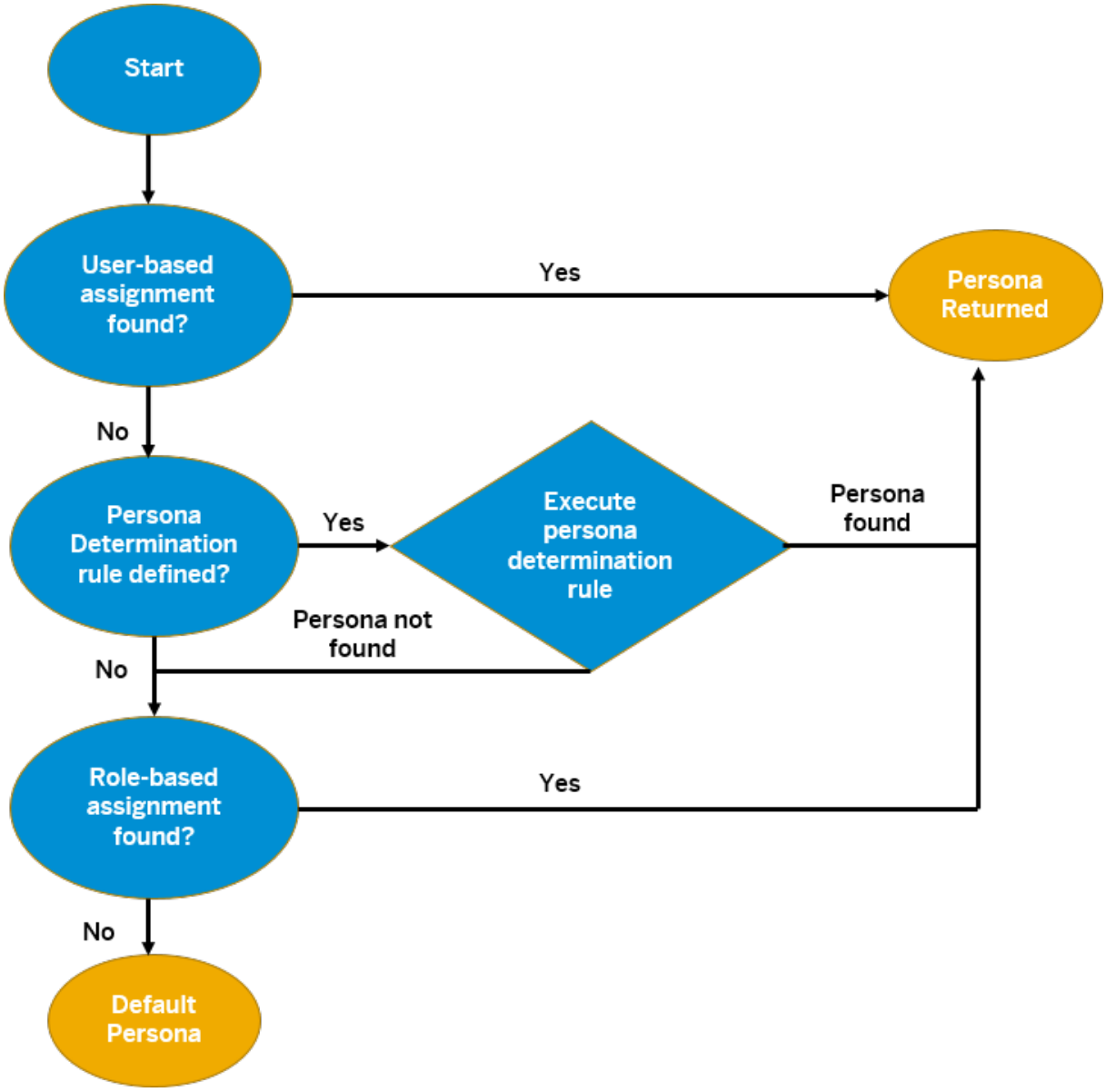
Application Type	Persona Type	Usage Type	Changeability
MAIF application	Maintenance Technician	Professional	No change allowed in customer system Changes allowed in SAP development system only
	Field Service Technician	Standard	
	Inventory Clerk	Standard	
	Custom	Professional	Change allowed in customer system. Custom persona can be created in customer namespace only.
Partner-defined MAIF application	Persona I	Professional	Can only be changed in partner development system
	Persona II	Standard	
	Persona III	Standard	
	Custom	Professional	Change allowed in customer system. Custom persona can be created in customer namespace only.
Custom-defined MAIF application	Custom	Professional	Change allowed in customer system. Custom persona can be created in customer namespace only.

3.1.3 Configuring Personas

SAP Service and Asset Manager allows a single user to be assigned multiple personas, and she may switch between them in the *Profile Settings* screen of the mobile application.

Context

The default persona determination rule, */SYCLO/CL_CORE_PERSONA_RULE*



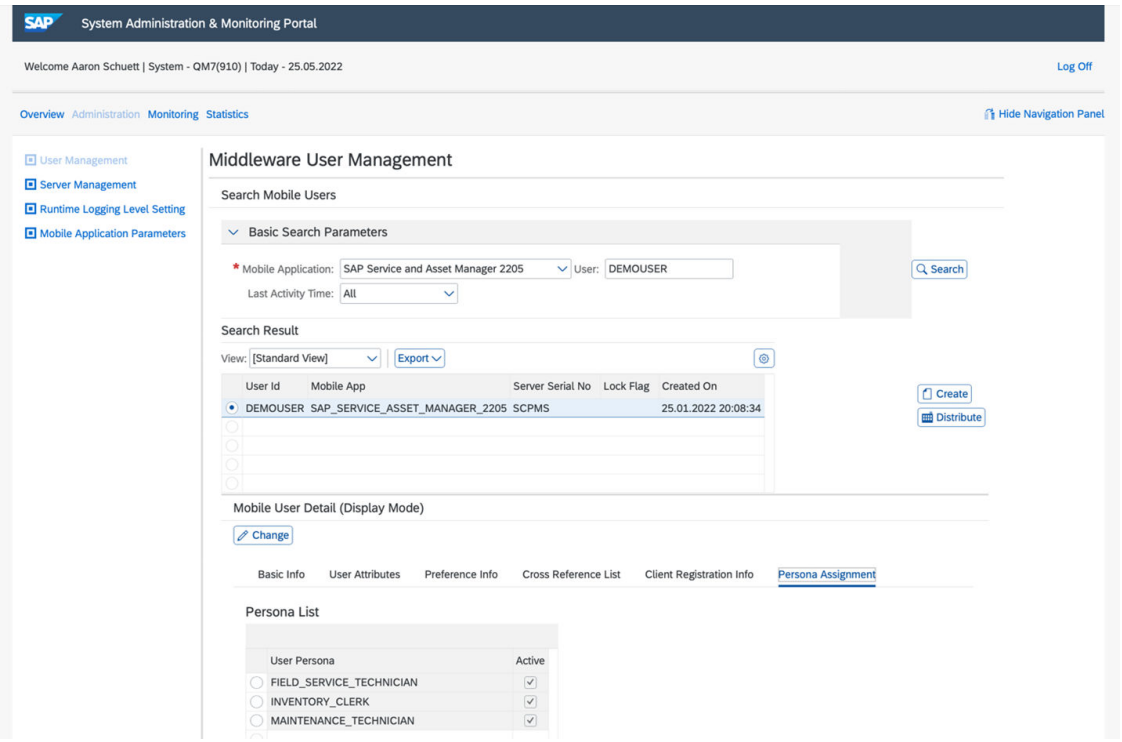
The system determines persona based on the following logic and sequence.

Procedure

1. Choose one of the following ways to assign a persona to a mobile user:
   a. **User-based assignment:** Navigate to *System Administration & Monitoring Portal* > *Administration* > *User Management*.

      Users may be assigned to one or more personas in the *Admin Portal*, launched in SAP GUI with tcode /n/syclo/admin.

      This is first considered to determine a user's persona. If a persona is assigned here, you will not need to do further checks.



      See the following topics and subtopics for detailed information, depending on your back end system:

      - Enabling Mobile-Specific Authorization Checks *Mobile Add-On for SAP S/4HANA Security Guide topic and subtopics in the* .
      - Topic and subtopics in the Enabling Mobile-Specific Authorization Checks topic and subtopics in the *Mobile Add-On for ERP Security Guide*.

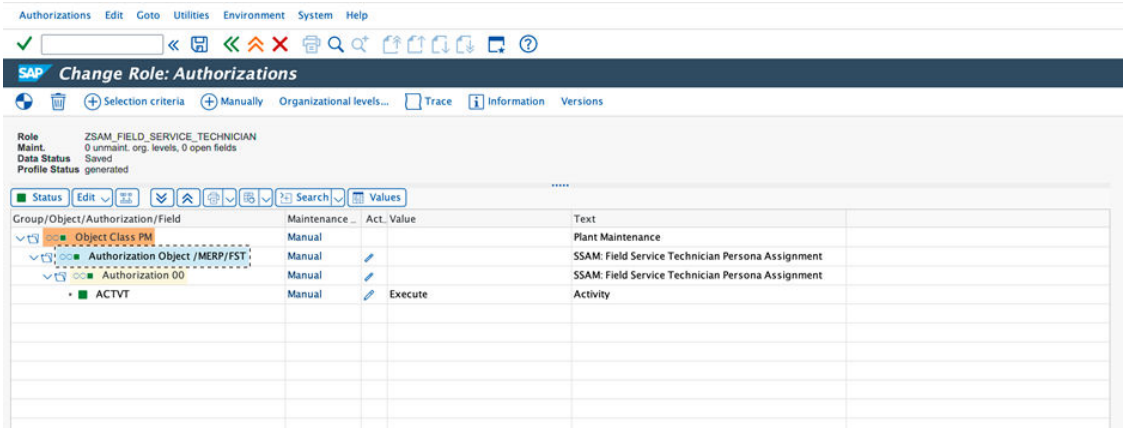
   b. **Role-based assignment:** Each persona is associated with a pre-defined authorization object in SAP Service and Asset Manager. Customers may assign the pre-defined authorization objects to users as needed. The authorization objects are then used to determine the user persona.

      These are typically assigned to a role in SAP GUI via t-code PFCG. By default, three standard authorization objects are delivered:

Persona	Authorization Object
Maintenance Technician	/MERP/SMT

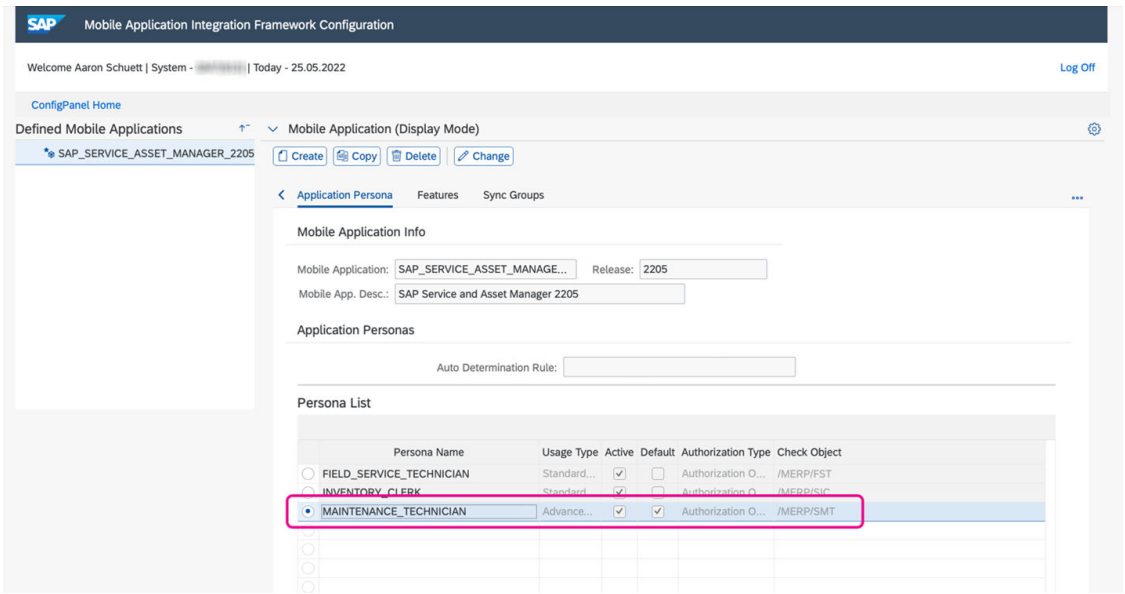
Persona	Authorization Object
Safety Technician	/MERP/WCM
Field Service Technician	/MERP/FST
Inventory Clerk	/MERP/SIC

Customers need to create custom role or roles and assigned authorization objects accordingly based on own business and security needs.



c. **Default persona assignment:** If none of the above options are found, the system will use the preconfigured default persona. The default persona is the *Maintenance Technician* in the standard delivery.

Customers can update the default persona based on own business needs by selecting the default checkbox in the *Configuration Panel*.



2. Implement your own custom rule, if desired. Select your custom rule in the *Auto Determination Rule* field.

3.1.4 Configuring Features

Customers can disable a feature if it is not relevant in their business scenarios, regardless of the personas used. Customers can also create features, not just enable or disable standard features.

Context

A new feature created by a customer can only be created in the customer namespace.

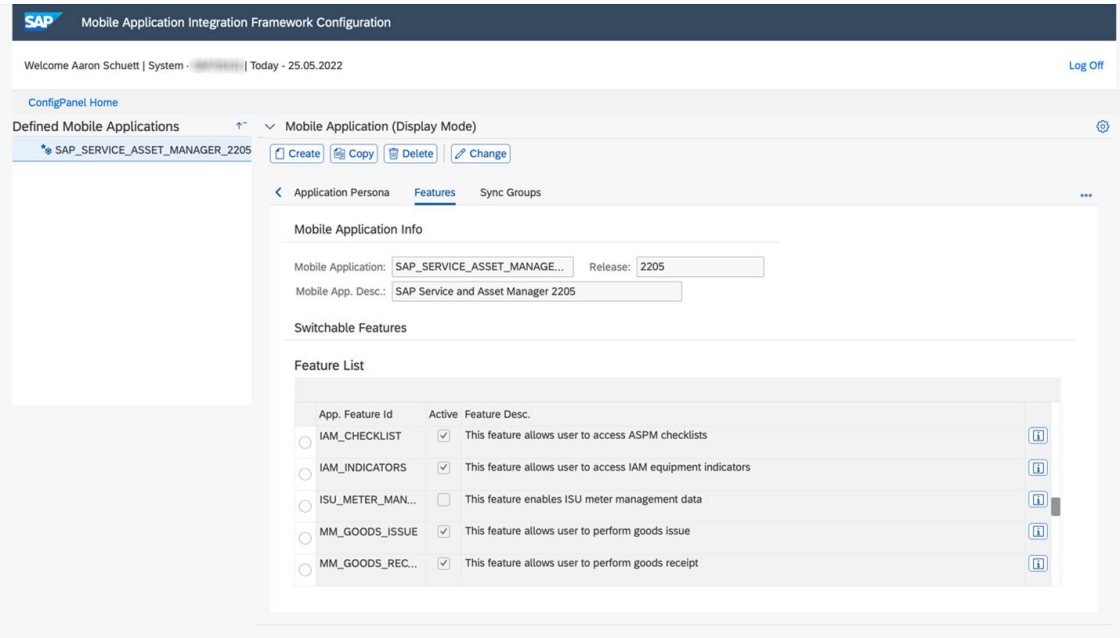
> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

Procedure

1. Choose one of the following ways to proceed:
   a. **Application level**: Navigate to *Configuration Panel* > *Mobile Application Configuration hyperlink* > *Select Mobile Application (SAP_SERVICE_ASSET_MANAGER_<version>)* > *Features tab* from the home page.

      The list of features delivered in SAP Service and Asset Manager is displayed in the *Configuration Panel*. You may define new features in a customer namespace only.

      Each feature has an *Active* check mark that you can toggle. This acts as the master switch across the application.

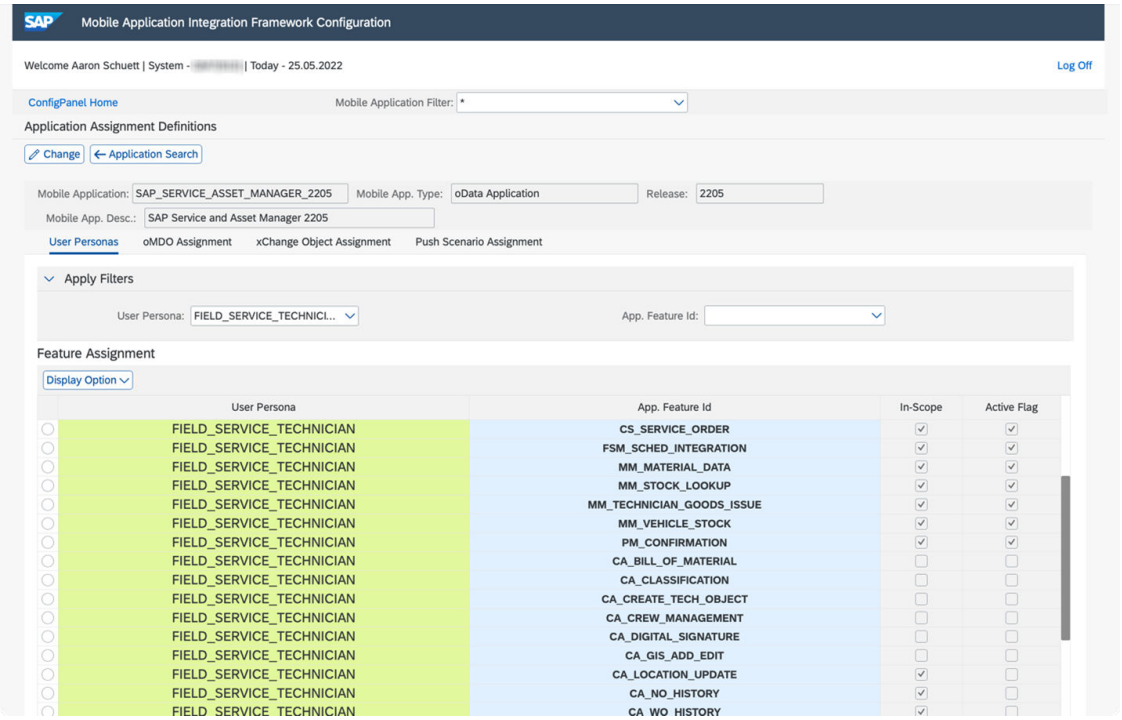


Note that enabling feature above does not check if system is qualified for the feature or not. If your system does not support ISU Meter Management, but you enable the ISU_METER_MANAGEMENT feature in the feature list, you could run into issues.

b. **Persona level**: Navigate to *Configuration Panel* > *Component Assignments hyperlink* > *Select Mobile Application (SAP_SERVICE_ASSET_MANAGER_<version>)* > *User Personas tab* from the home page.

   You can also toggle features related to a persona. Each persona is configured with a list of features supported in the standard delivery of SAP Service and Asset Manager.

   You can view a list of features by persona in the *Configuration Panel*. Each feature has an *In-Scope* checkbox, meaning that the feature is allowed for that persona. In-scope features are not editable in customer or QA environments. Only In-Scope items can be enabled or disabled using the *Active Flag* checkbox.



   You can enable a feature for one persona, but disable the same feature for another persona. The list can be filtered using the *User Persona* or *Application Feature* Id drop-down list to find entries directly.

   Note that if the feature is disabled at the application-main switch level, the feature is still considered disabled, regardless of the *Active Flag* status at the persona level.

2. Click the *Change* button. Enable or disable desired features.
3. *Save* your changes.

Results

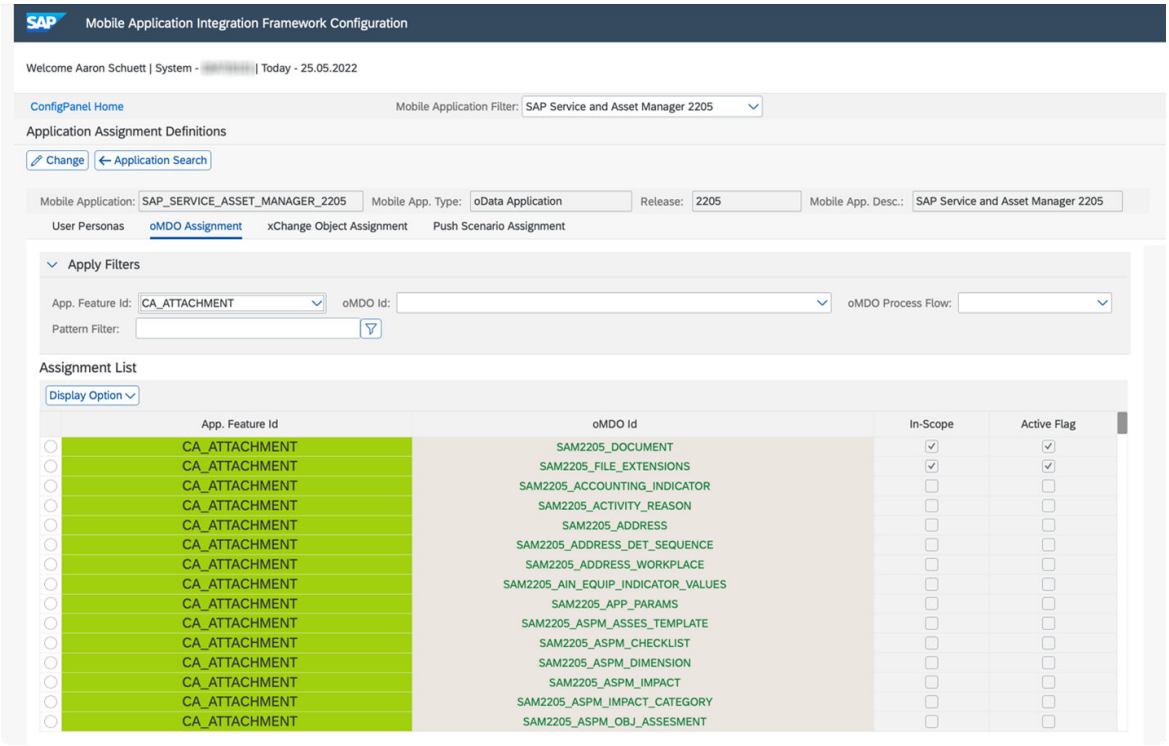
Enabled features display in the *Component Assignment* > *Switchable Features* tab.

3.1.4.1 Performance Calibration

During mobile client synchronization, all active data objects are sent to the mobile user. For example, if data is being sent to the mobile user that the client application does not need, this can increase the synchronization time.

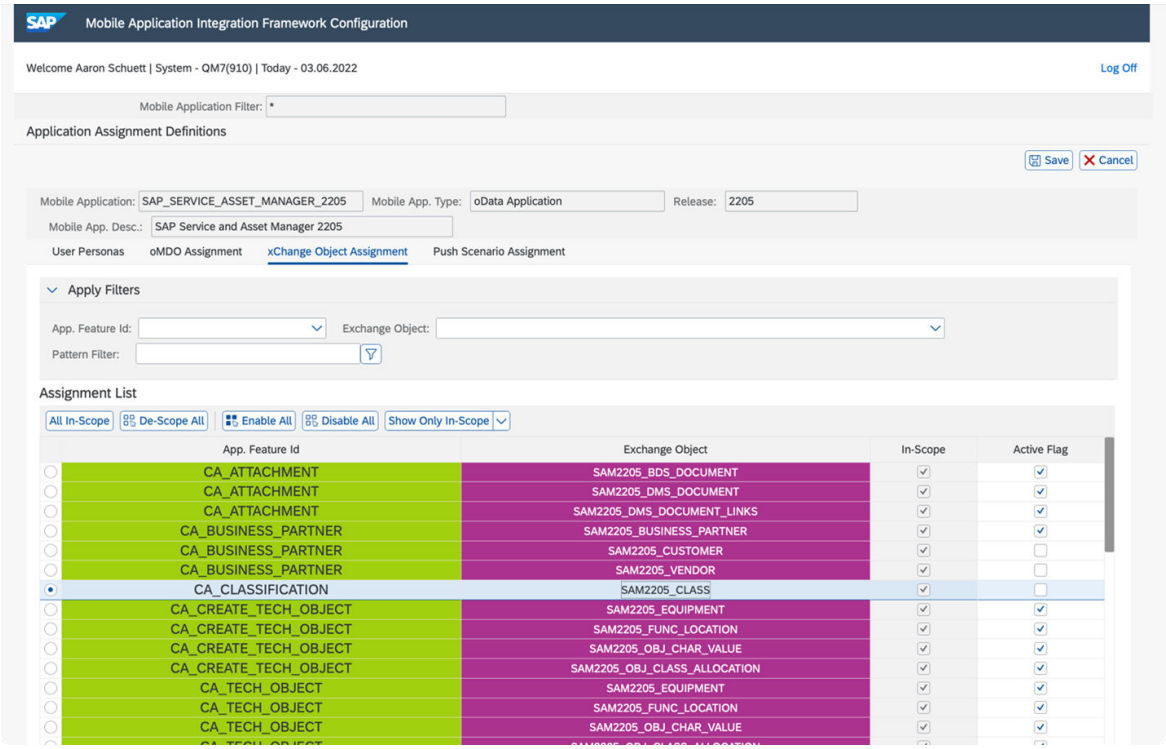
Disable the data objects that are not used in your workflow. Below are several ways to accomplish this.

- **oMDO Assignment**: Navigate to *Configuration Panel* > *Component Assignments hyperlink* > *Select Mobile Application (SAP_SERVICE_ASSET_MANAGER_2205)* > *oMDO Assignment tab*.
  Each feature is associated with one or more oData Mobile Data Object (oMDO) which provides business logic for a business object used in the mobile application and is mapped to OData requests for a business object on the backend.
  You can view an oMDO list in the *Configuration Panel* and choose which entities are required and disable the others to avoid requesting unnecessary data during the mobile sync.

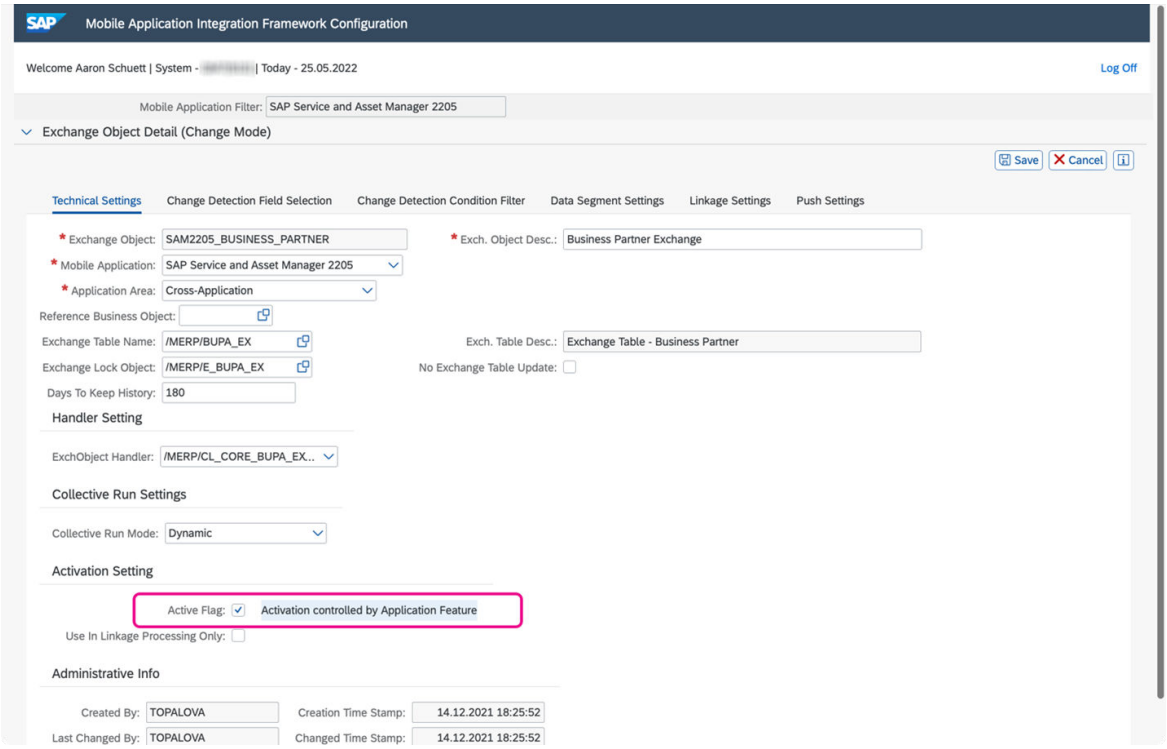


  Like feature assignments, the *In-Scope* check mark indicates the data objects that are supported for each feature. Only In-Scope items can be enabled/disabled using the *Active Flag* check mark, and In-scope features are not editable in customer or QA systems.

- **Exchange Object Assignment:** Navigate to *Configuration Panel* > *Component Assignments hyperlink* > *Select Mobile Application (SAP_SERVICE_ASSET_MANAGER_2205)* > *xChange Object Assignment tab*.
  Features can also be associated with an exchange object that combines technical objects, such as the exchange class handler, the exchange table and the lock object, with configuration rules.
  You can view the list of exchange objects in the *Configuration Panel* and select the ones you need and disable the others.



You can also modify the active objects individually using the exchange object configuration. In this case, navigate to *Configuration Panel* > *Exchange Object Configuration hyperlink* > *Mobile Application Filter (SAP_SERVICE_ASSET_MANAGER_2205)* > *Exchange Object By Application Area* > *Technical Settings tab* >.



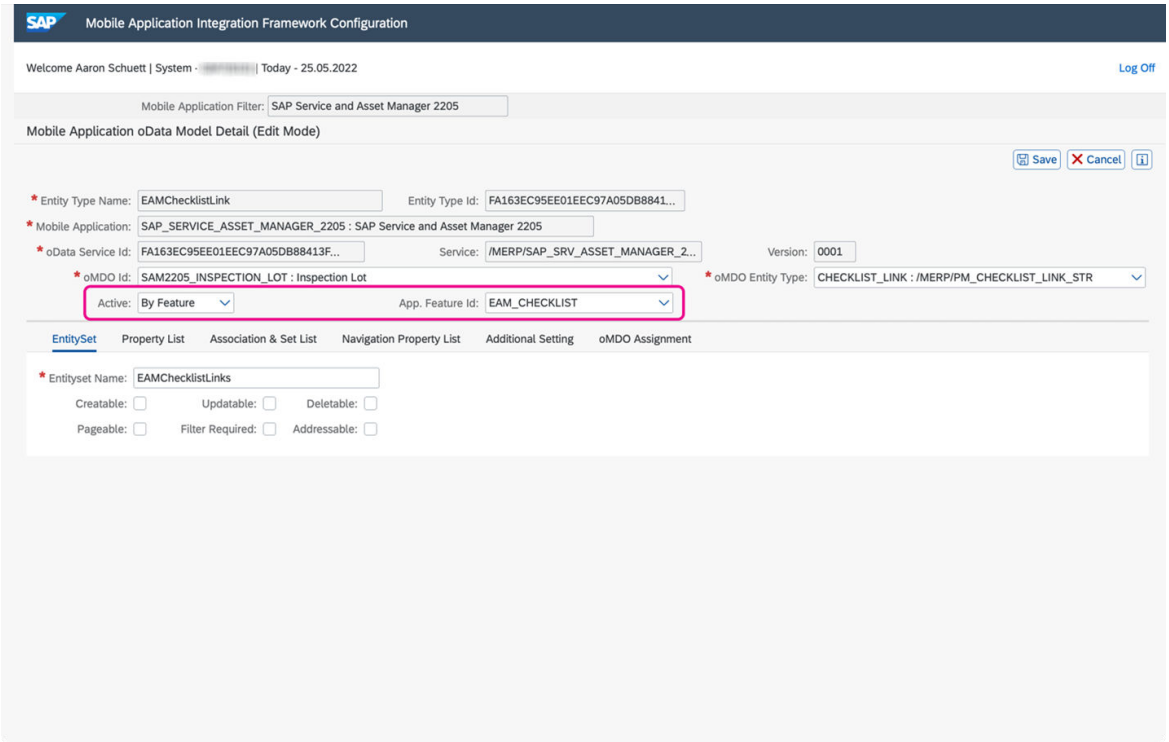
If the exchange object is assigned to a feature, the check mark *Activation controlled by Application Feature* will appear next to the *Active* check mark.

> **Note**
>
> This control is like a master switch; if activation is disabled here, it will be disabled in all places, regardless of the active flags set in the previously discussed methods. However, it is also important to note that if an object is active here but disabled by the other methods, the linked exchange will not be executed when changes are made to the object. This prevents unnecessary exchanges from being triggered if the feature is not applicable to customers.

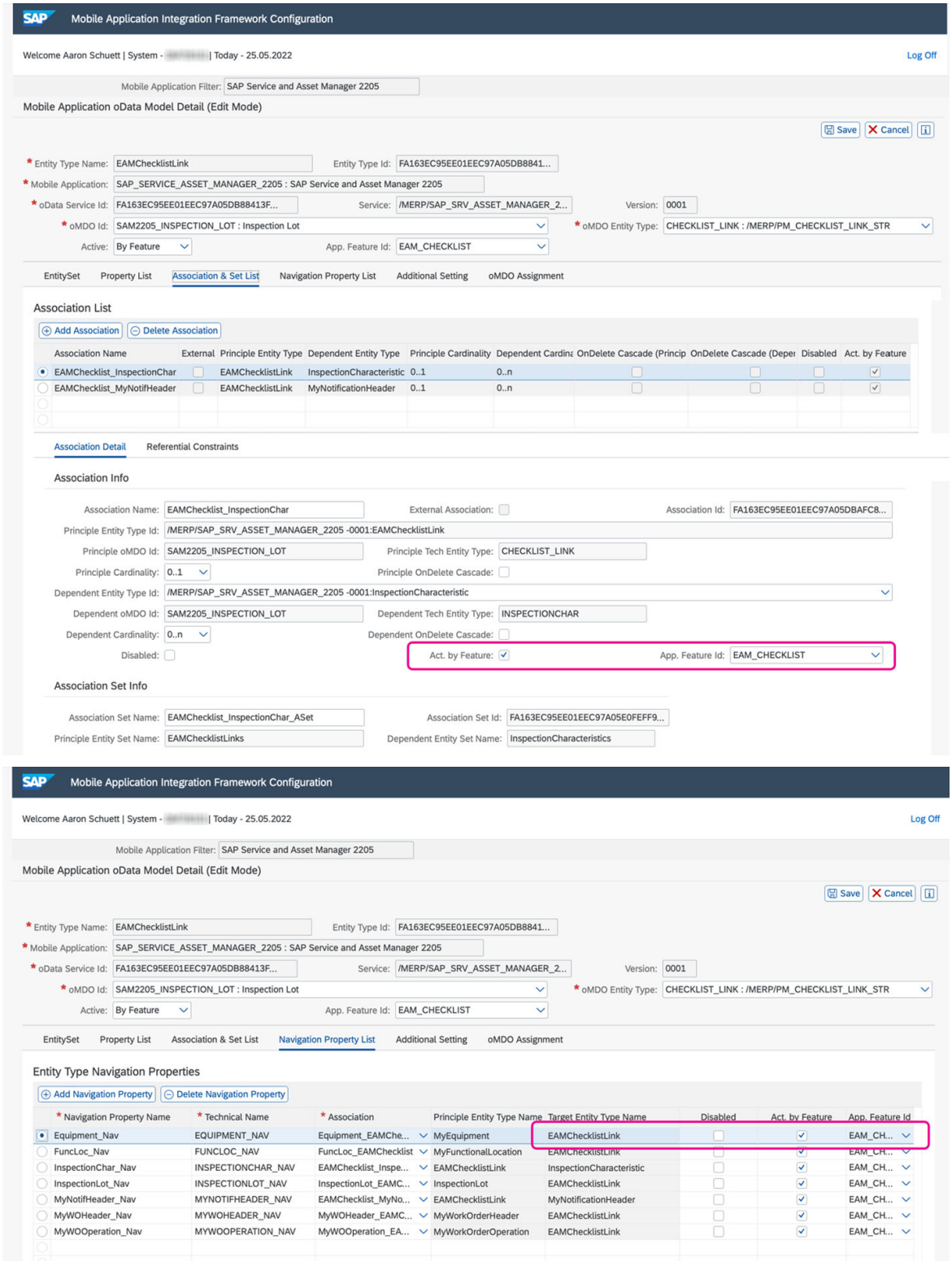
3.1.4.1.1 Activation by Feature

There are several entities in the Configuration Panel where it is possible to support activation by feature. This allows you to go down the hierarchy to manage entities, dependency objects, and more.

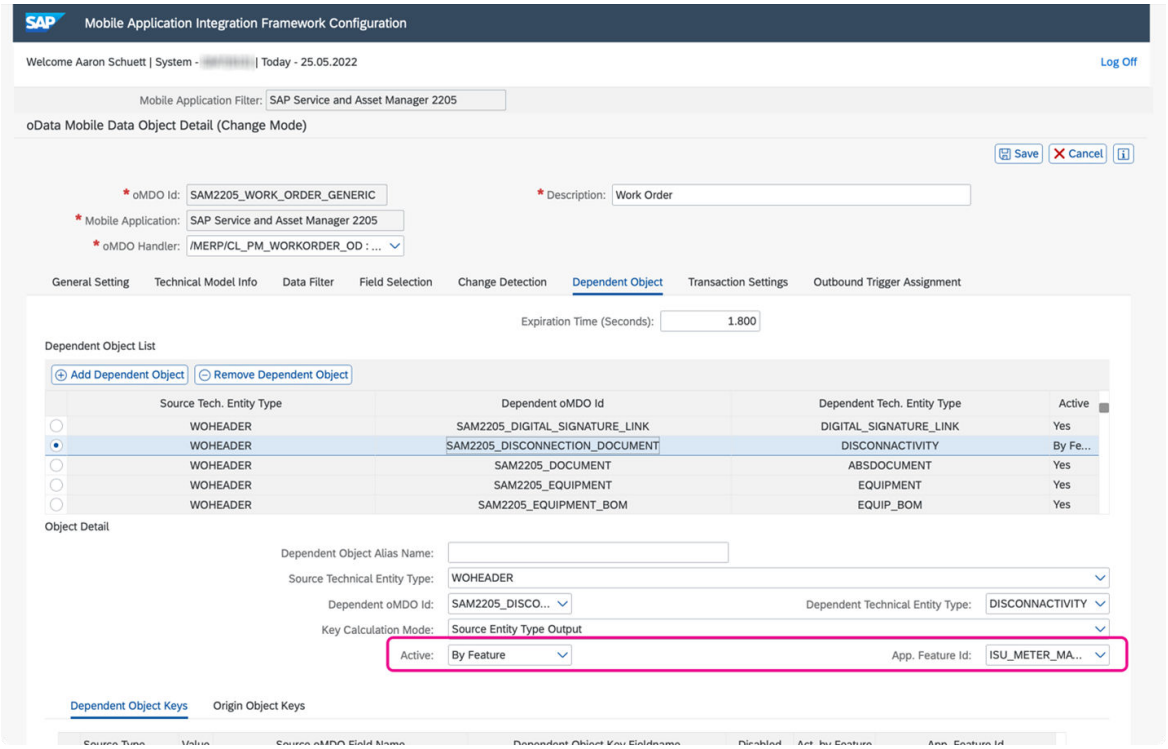
- **oData Model Entity Type:** You can specify a particular entity to be enabled only if the assigned feature is activated. In the example below, the EAMChecklistLink entity set is applicable to the Enterprise Asset Management (EAM) checklist functionality. Here only when EAM_CHECKLIST feature is enabled, EAMChecklistLink is available for the client.



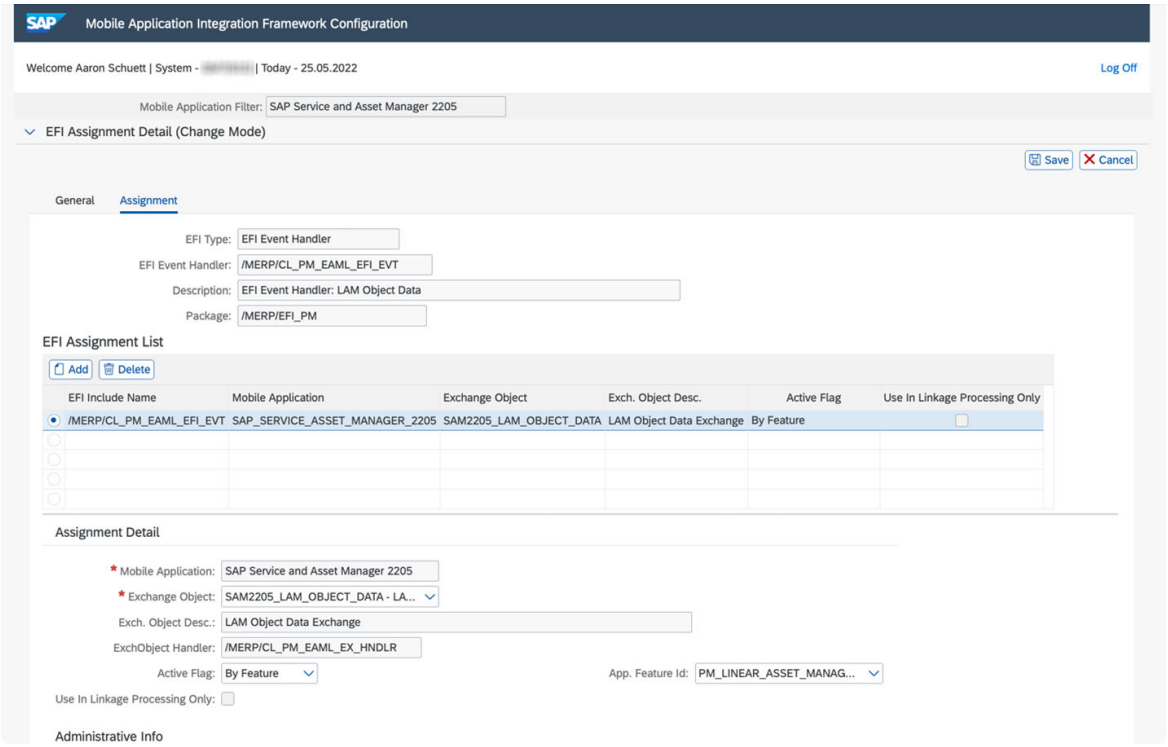
- **Association and navigation:** Like an entity type, association and navigation can be assigned to a feature. Only when the assigned feature is enabled does the association become visible to the mobile client. If the disabled flag is active, the entity will not be visible to the mobile client regardless of whether the feature is enabled or not.



- **Dependency Object:** If activation by feature is configured in a dependent object, and if the assigned feature is not enabled, dependency objects will not be generated in the dependency queue.



- **EFI configuration:** An Enhancement Framework Implementation (EFI) event handler can be assigned to a feature with active flag set to *By Feature*.



> **Note**
>
> We only support one feature assignment per object (such as entity type, association, navigation, EFI). You cannot assign multiple features to an object in the Configuration Panel.

3.1.5 Configuring Component Assignments

The *Component Assignment* page allows you to define what features are applicable for each persona.

Context

Using the *User Personas* tab, you can enable or disable features as desired. For example, *PM_CONFIRMATION* is enabled by default, and *HR_TIMESHEET* is disabled. If your site uses CATS, you can disable PM_CONFIRMATION and enable HR_TIMESHEET.

The *Switchable Features* tab allows you to configure OMDOs required for each feature. During initial and delta syncs, the client only downloads data from the assigned OMDOs. Activating specific OMDOs determines the entities and data that is downloaded to the mobile device.

Procedure

1. Navigate to *Component Assignments* from the home page. Select your application in the *Mobile Application Filter* dropdown. Click on your application hyperlink in the *Search Result* table.
2. Click the *Change* button. Enable or disable desired features using the *Active Flag* checkbox. *Save* your changes.
3. Select the *Switchable Features* tab.
4. Click the *Change* button. Enable or disable desired OMDOs using the *Active Flag* checkbox. *Save* your changes.

> **Note**
>
> Don't select OMDOs that belong to the online service (ex: */MERP/SAP_ONLINE_LOOKUP_EXT_<version>*). These online service entities might not exist in the base service (*/MERP/SAP_ASSET_MANAGER_<version>*).

> **Note**
>
> If you don't see an expected feature in the list, ensure it's enabled on the *Mobile Application Configuration* > *Features tab*. See the Configuring Features [page 90] procedure for more information.

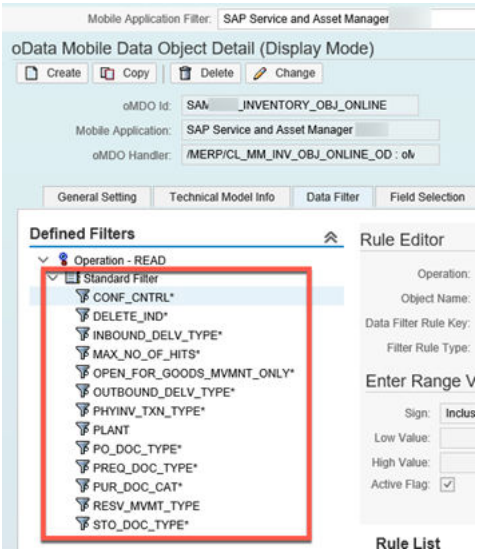
Next Steps

After you've enabled or disabled a feature in the *Component Assignments* page, navigate to *Home* > *Mobile Application Configuration* > *Parameters tab*. Find the parameter group and parameter name and maintain the feature enablement accordingly.

3.1.6 Inventory Clerk Parameter Configuration

You can search for inventory objects you want to process directly from your mobile device.

Based on the search criteria, the application will search for objects locally or in the backend (if enabled). Objects matching the oMDO filter values will be returned in the search results.



- CONF_CNTRL: Default value is blank and used in PO and STO search
- DELETE_IND: Default value is blank and used in PO, STO, RS, and PI search
- INBOUND_DELV_TYPE: Default value is ‘EL’ and used in IB search
- MAX_NO_OF_HITS: Default value is ‘1000’ and applied for each object search individually
- OPEN_FOR_GOODS_MVMNT_ONLY: Default value ‘X’ and applied for in each object search

  > **Note**
  >
  > This filter is cascaded and set to active for all inventory objects oMDO except the physical inventory.

- OUTBOUND_DELV_TYPE: Default value is ‘LF’ and used in OB search
- PHYINV_TXN_TYPE: Default values are ‘IB’ and ‘IN’. Used only in the physical inventory search.
- PLANT: Default value is none and used in each object search
- PO_DOC_TYPE: Default value is ‘NB’ and used in PO search
- PUR_DOC_CAT: Default value ‘F’ and used in PO search
- RESV_MVMT_TYPE: Default value is none and used in RS search

- STO_DOC_TYPE: Default value is ‘UB’ and used in STO search.

3.2 Mobile Application Configuration Procedures

3.2.1 Mapping Work Orders to a STARTED Status - Overview

By default, the SAP Service and Asset Manager application maps the *STARTED* work order status on the client to the *REL* status in SAP Mobile Add-On.

In many implementations, a status of *MOBI* is used in SAP Mobile Add-On to indicate that the work order is started by a technician. The *MOBI* status cannot be modified on the back end.

You can map the mobile status to a different status within SAP Mobile Add-On by altering the mobile application configuration for SAP Service and Asset Manager and changing the system status technical code for the STARTED mobile status. After you change the system status technical code, updates to SAP Mobile Add-On made when a user starts a work order set the status in SAP Mobile Add-On to the MoBI status, matching the entered technical code.

The only modification to make is in the ConfigPanel, in the *Mobile Application Configuration* page, *Mobile Status Setting* tab. Change the mobile status for a started work order in the list of the mobile status options for SAP Service and Asset Manager, with the system status value of that same record altered to use the technical code of the desired status.

3.2.1.1 Changing the Mapping of a Mobile Status to STARTED

Prerequisites

Address the following before performing this procedure:

- Determine and note the technical code of the work order system status to which the mobile status STARTED will be mapped, as it is used in the procedure.
- The system status to which you are mapping the mobile status of STARTED in this procedure is configured as a work order status.
- The person performing this procedure has access to the ConfigPanel and permissions to change configuration settings of the elements within it.

Context

The following procedure describes the steps required to change a system status when a mobile STARTED status is mapped to it.

Procedure

1. Starting from the ConfigPanel home page, click the *Mobile Application Configuration* link. Then click the *Mobile Status Setting* tab.
2. Choose your desired mobile application from the list of *Defined Mobile Applications* in the left pane.

   The application level status settings display in the tab to the right. Information includes the *Mobile Status List*.
3. In the *Mobile Status List* table, find the *Object Type* of `<WORKORDER>` with a *Mobile Status* of `<STARTED>` and click the *Change* button.
4. Change the *System Status* value to the technical code of the system status to which the STARTED mobile status should be mapped. When done, click *Save*.

Results

After completion of the procedure, the STARTED mobile work order status is mapped to a different system status than the default REL status.

3.2.2 Configuring Auto-Sync

You can enable, disable, and customize parameters related to the auto-syncing of the app in the ConfigPanel.

Context

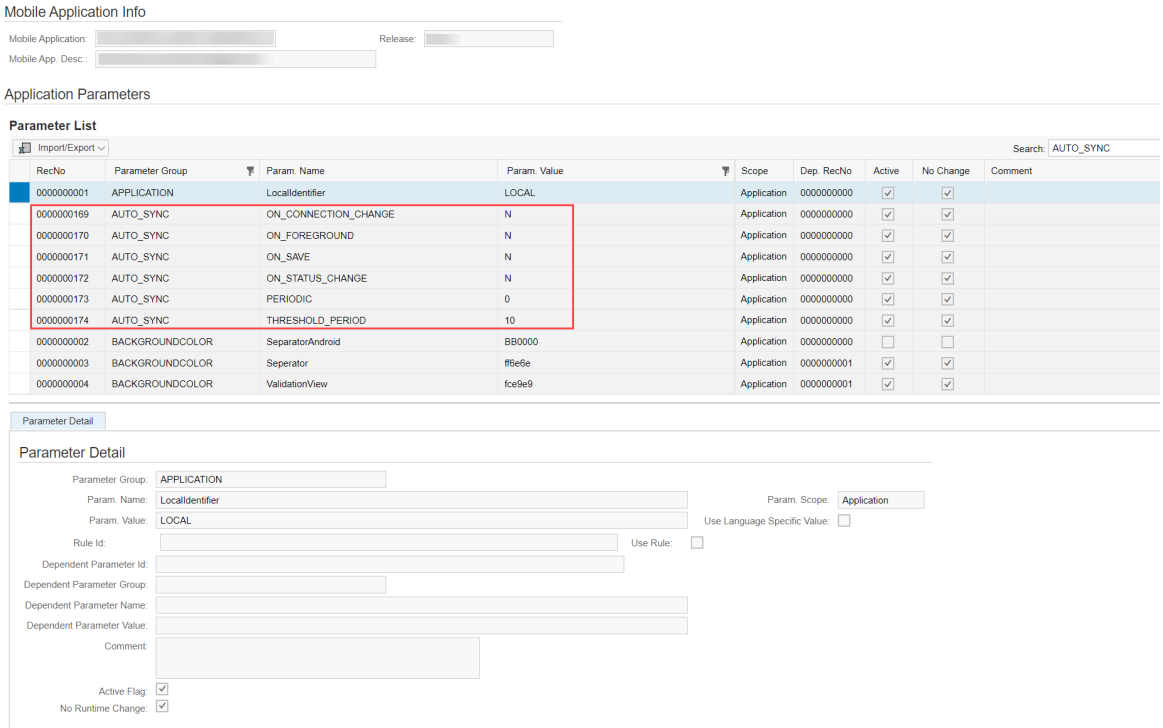
You can configure the following parameters to customize the auto-syncing of the app in the *AUTO_SYNC* parameter group:

- *ON_CONNECTION_CHANGE*: Enables or disables auto-sync on network connectivity changes
- *ON_STATUS_CHANGE*: Enables or disables auto-sync on mobile status changes
- *PERIODIC*: Enables or disables periodic auto-sync
- *ON_FOREGROUND*: Enables or disables auto-sync when the app moves to the foreground on the mobile device
- *ON_SAVE*: Enables or disables auto-sync on a database save. Used in conjunction with the *THRESHOLD_PERIOD* parameter.
- *THRESHOLD_PERIOD*: If enabling the *ON_SAVE* parameter, set the threshold period in minutes.

Procedure

1. Ensure the *CA_AUTO_SYNC* feature is enabled for your personas. See the Configuring Personas and Features Overview [page 83] topic and subtopics for detailed information.
2. Navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.
3. Configure one or more parameters in the *AUTO_SYNC* group as desired.



4. Check the *<Active>* flag to ensure that the parameter is used by the mobile application. If desired, and if not already checked, check the *<No Runtime Change>* box to ensure the value of the parameter isn't overridden at runtime through synchronization processing.
5. *Save* your changes.

3.2.3 Configuring CATS and PM Confirmation Time Entries

Use parameters to configure CATS and PM confirmation minute interval values.

Prerequisites

If you're using CATS, the SAP HR module must be installed.

Context

The HR Timesheet feature is called *HR_TIMESHEET*. The PM Confirmation feature is called *PM_CONFIRMATION*. By default, the PM Confirmation feature is enabled. Note that only one feature should be enabled at any given time.

> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

Use the *CatsMinuteInterval* parameter when CATS is enabled and the *LaborTimeMinutesInterval* parameter when PM confirmations are enabled. The following procedure is the same for either parameter, even though this guide is using the *CATSMinuteInterval* parameter as an example.

When a mobile user manually logs their time, or their time is automatically logged for them through the use of the application, the time logged is rounded to the nearest interval configured. For example, you manually log an additional 12 minutes of work on a work order on a mobile device. Your *CATSMinuteInterval* parameter is set to *15*. Therefore, your additional time logged is automatically rounded up to 15 minutes. The time entry screens also have their duration control values limited to minute values matching the configured interval.

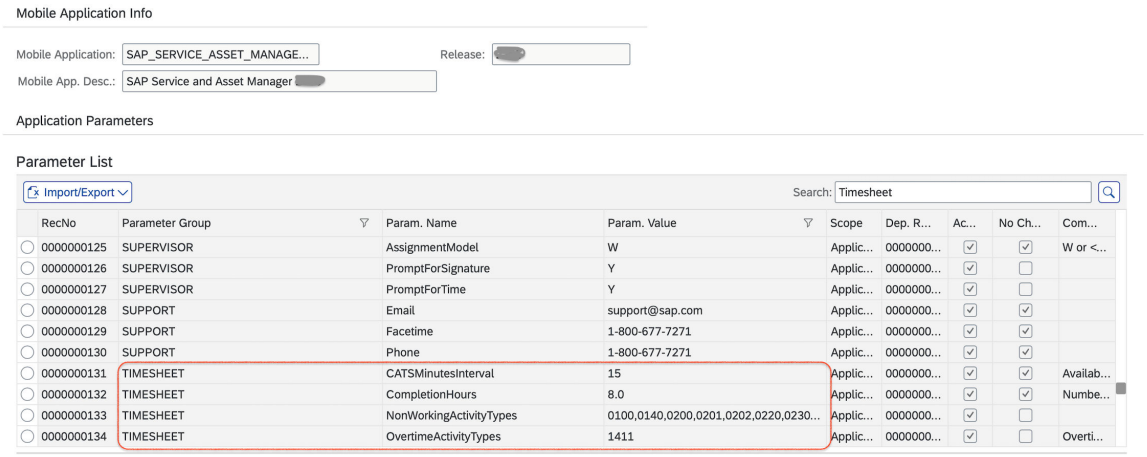
The default value for both parameters is *15*.

Procedure

1. Using the ConfigPanel, navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.
2. The *CATSMinuteInterval* parameter is found in the *TIMESHEET* group. You can scroll down to find the parameter, or perform a search using the *Search* box. Highlight the *CATSMinuteInterval* parameter and click the *Change* button.

   > **Note**
   >
   > You can find *LaborTimeMinutesInterval* parameter in the *PMCONFIRMATION* group.



3. You can change either the CATS or the PM confirmation labor time interval to the following values:
   - 1
   - 5
   - 10
   - 15
   - 30

   If you accidentally set the parameter to an interval value that isn't an allowed value, the parameter automatically defaults to a value of *15* on the client device.
4. Check the `<Active>` flag to ensure that the parameter is used by the mobile application. If desired, and if not already checked, check the `<No Runtime Change>` box to ensure that the value of the parameter isn't overridden at runtime through synchronization processing.
5. *Save* your changes.

3.2.4 Configuring Clock In Clock Out

The Clock In Clock Out (CICO) feature decouples time tracking from the mobile status of a work order or operation, allowing multiple users to start and log time against the same work order or operation simultaneously.

Overview

> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

The Clock In Clock Out feature is called *PM_CLOCK_IN_CLOCK_OUT* .

The CICO parameter in the ConfigPanel allows multiple users to work on the same work order or operation, where all users receive the work orders and operations to their devices. CICO allows multiple work orders and operations to be in the Started state that belong to different users. Mobile device users are able to see the CICO state of all work orders or operations on their device. Users can also filter their work order or operations lists based on clock in or clock out status.

If CICO is enabled in the ConfigPanel:

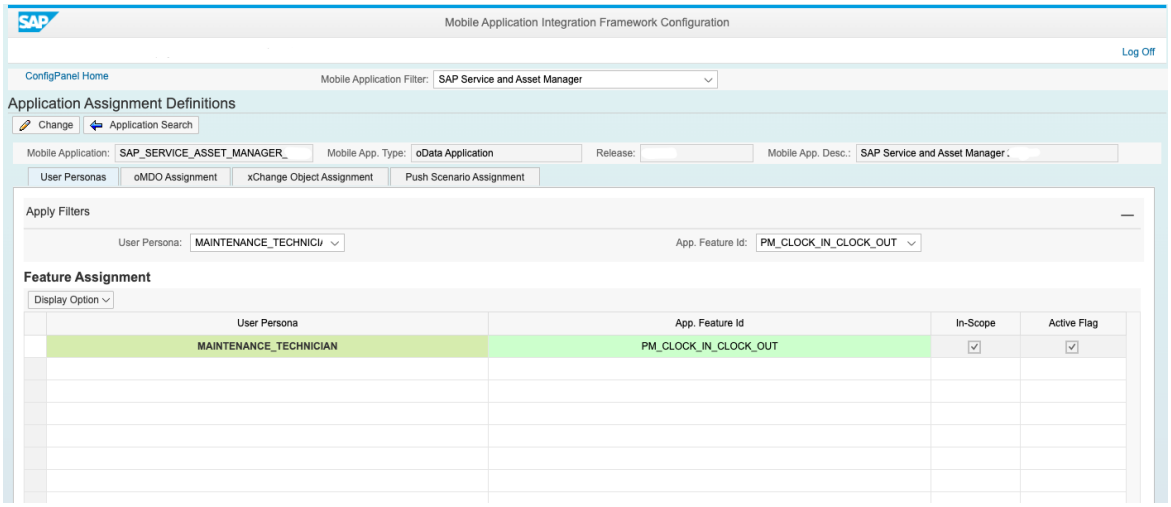
- Multiple people can work on the same work order or operation even if the work order or operation is already started by another user
- Users can clock in to any work order or operation on their device
- Users can only clock in to one work order or operation on their device at a time
- Users must clock out of the current work order or operation before clocking in to a different work order or operation
- All time recording (CATS and Confirmation) uses the clock in clock out period as the default duration in time entry screens
- When a user clocks in to a work order or operation:
  - The timestamp of the work order or operation is saved to a user-specific table that is persisted in the back end
  - The mobile status of the work order or operation is set to Started if it isn't already in a started state
- When a user clocks out of a work order or operation:
  - The work order or operation status is set to either Hold or Complete
  - If a work order or operation is set to Complete and Confirmations are used, the user can set it as the final confirmation

If CICO is disabled in the ConfigPanel:

- A user can start any work order or operation that is in a Hold, Received, or Local state
- A user can start only one work order or operation at a time
- The mobile status of a work order or operation is used to track time in either CATS or Confirmations

How to Set the CICO Parameter Using the Configuration Panel

1. Using the ConfigPanel, navigate to *Component Assignments*. Select your application in the *Mobile Application Filter*. Click the application link in the *Search Result* table.
   The *Application Assignment Definitions* page displays.
2. Click the *Change* button. Find the PM_CLOCK_IN_CLOCK_OUT feature ID in the MAINTENANCE_TECHNICIAN user persona part of the table. Enable or disable the feature using the *Active Flag* checkbox.



3. *Save* your changes.

3.2.5 Configuring Signature Control

Use parameters to configure signature control.

Context

Use the *SIGN_CAPTURE* parameter group and the following parameters within the group to configure signature control for SAP Service and Asset Manager:

- OP.Complete: Enable for operations
- SubOp.Complete: Enable for suboperations
- WO.Complete: Enable for work orders

> **⚠ Restriction**
>
> The digital signature feature is available in SAP S/4HANA 2020 and above releases. The digital signature feature is not available for SAP ERP systems or SAP S/4HANA systems lower than 2020.

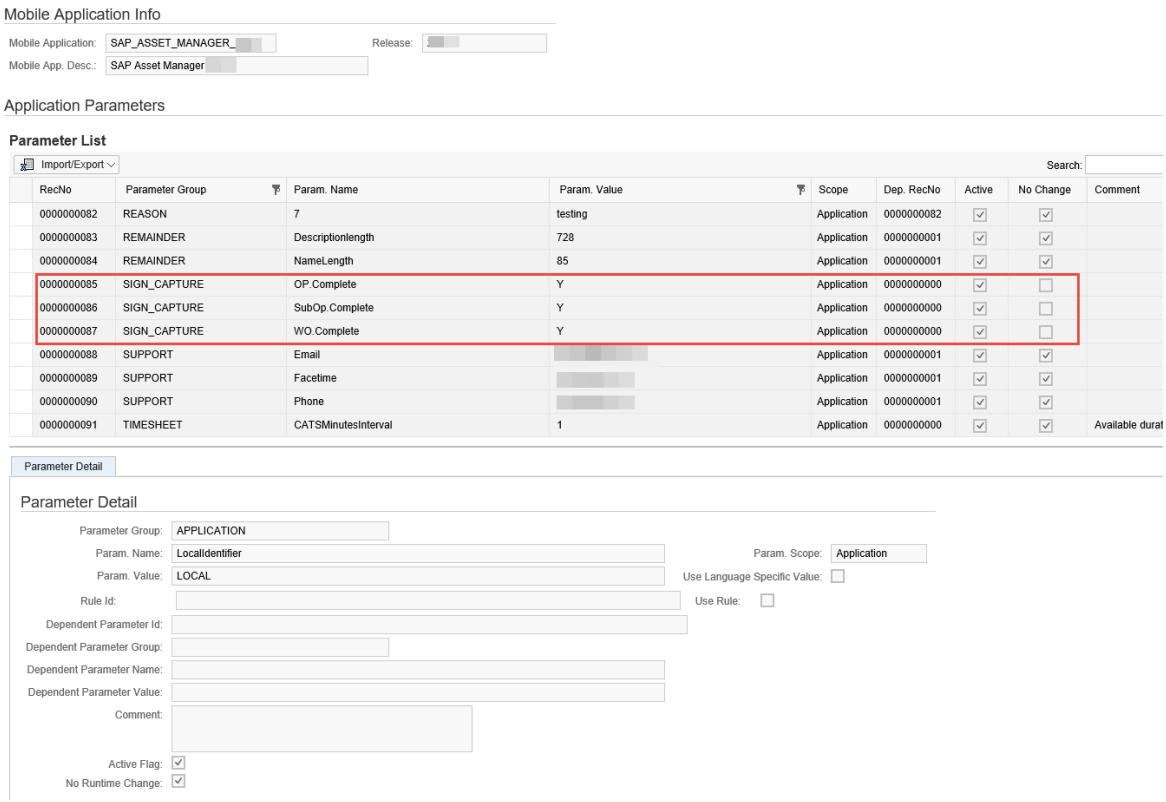
By default, the signature control feature is not enabled. To enable signature control:

Procedure

1. Using the ConfigPanel, navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.

2. The signature control parameters are found in the *SIGN_CAPTURE* group. You can scroll down to find the parameter, or perform a search using the *Search* box. Highlight the parameter you want to configure and click the *Change* button.



3. Make your desired parameter association changes. Enable the parameter as follows:

   - **Y:** Control is displayed on the client and is required before completing the object
   - **N:** Control is not displayed on the client (enabled by default)
   - **O:** Control is displayed on the client and is an optional step to complete the object

   > ⓘ Note
   >
   > If the parameters have no *Parameter Value* assigned, signature control does not display on the client.

4. Check the `<Active>` flag to ensure that the parameter is used by the mobile application. If desired, and if not already checked, check the `<No Runtime Change>` box to ensure that the value of the parameter is not overridden at runtime through synchronization processing.
5. *Save* your changes.

3.2.6 Configuring Namespace Check

SAP provides namespace check for the following configuration areas:

- oMDO class
- oData Model Entity Type Name

- Entity Set Name
- Entity Type Property Name
- Association Name
- Association Set Name
- Navigation Property Name
- Navigation Property Technical Name
- User Attribute

This works so that only Y or Z namespace can be used in the user SAP system while only the partner namespace is allowed in the partner system.

With SAP Service and Asset Manager 2210 or earlier versions, you can bypass namespace checks by setting the *NO_NAMESPACE_CHECK* runtime parameter to X. Namespace bypass is not applicable to the Persona and Feature configuration, whereby a custom or partner namespace remains applicable in those areas and cannot be bypassed.

> **Note**
>
> As of SAP Service and Asset Manager 2305, you can no longer use the runtime parameter to bypass namespace check. A custom or partner namespace is enforced in the above mentioned configuration areas. For more information on how to set the runtime parameter, see the https://launchpad.support.sap.com/#/notes/2713969 SAP note.

3.2.7 Digital Signature Overview

The digital signature feature performs a verification of work by digitally signing for the work using a user name and password. The digital signature framework is able to create digital signature verification by using a back-end user name and registered time-based one-time password, or TOTP. The TOTP is brought by the CL_TOTP object.

The digital signature UI5 reuse component provides an OData service that allows for direct online interaction by the client to register a digital signature completed work.

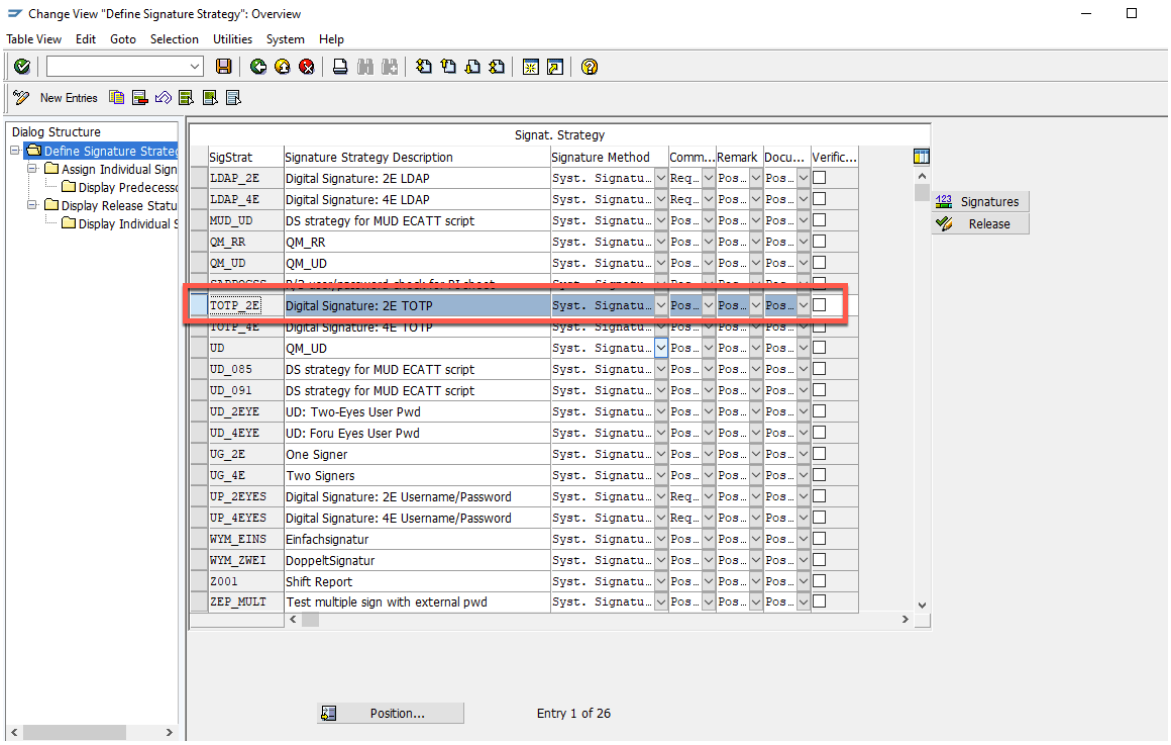
The Digital Signature feature is called *CA_DIGITAL_SIGNATURE*. By default, the feature is not enabled.

> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

For complete information on digital signature, see the Digital Signature guide, specifically the Implementation Guide for UI5 Reuse Component topic and subtopics.

Limitations

- SAP Service and Asset Manager digital signature relies on the CL_TOTP object interface.
- SAP Service and Asset Manager digital signature uses a connection to the UI5 Reuse component in digital signature. The UI5 Reuse component provides the ability to digitally sign different objects.
- Digital signature support from SAP Service and Asset Manager is only available on an SAP S/4HANA 2020 or later back end system. Digital signature is therefore available for SAP Service and Asset Manager 2105 or later releases installed on a SAP S/4HANA 2020 or later back end system.
- Digital signature is supported with any two-eyed (2E) strategy. Out of the box, the TOTP_2E strategy is used. TOTP_2E is the two-eyed principle of a technician signing the operation for their work. You can configure the 2E strategy by navigating to *OData Mobile Data Object Configuration* > *Data Filter*. Select the **SAM2310_DSIG_CONFIGURATION** OMDO with the filter *SIGNATURE_STRATEGY*.



3.2.7.1 Enabling Digital Signature

Prerequisites

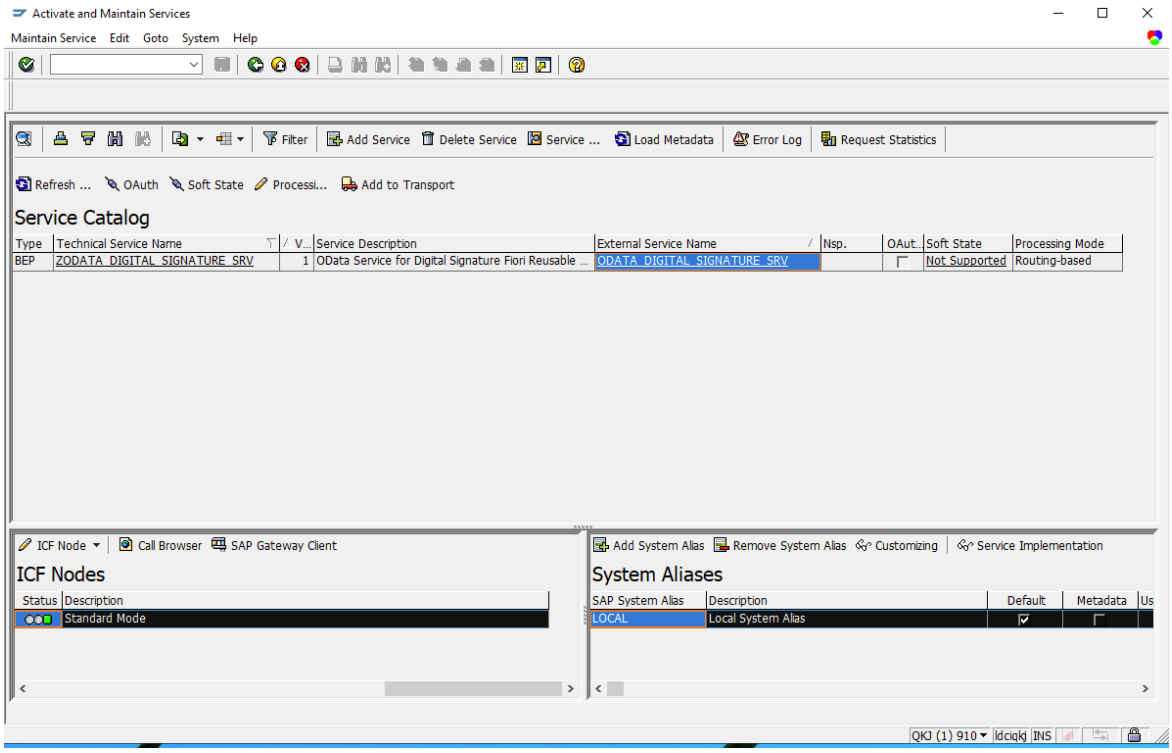
Ensure you're following all points discussed in the *Limitations* section of the Digital Signature Overview [page 107] topic.

An authenticator application, such as Microsoft Authenticator, must be installed on the mobile device, in order to use the TOTP process.

> **Note**
>
> To enable digital signature for work order headers, follow the tutotial https://blogs.sap.com/2021/11/09/sap-asset-manager-enable-digital-signature-for-work-order-headers/.

Procedure

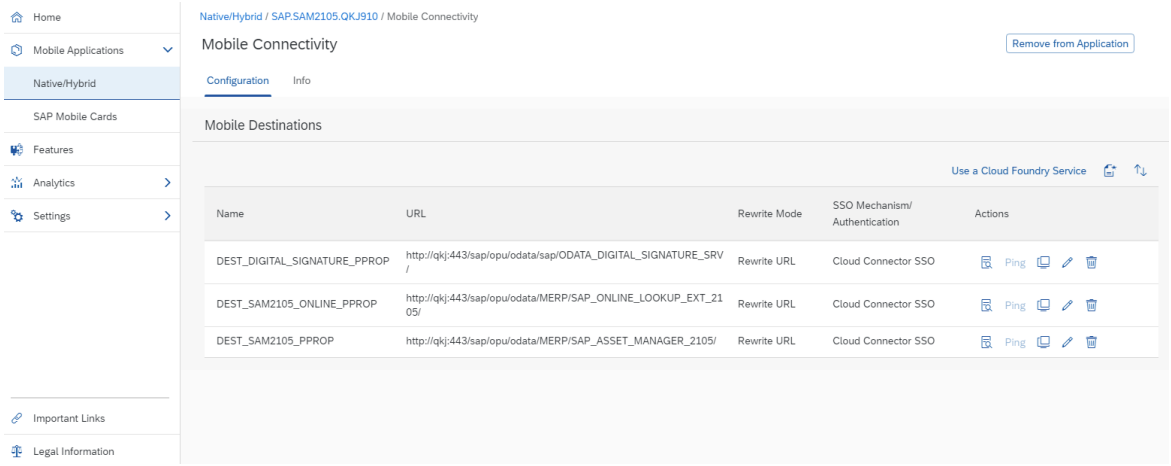
1. **Create a destination in SAP Business Technology Platform Mobile Services (SAP BTP services)**: Create a specific destination in SAP BTP services that points to the digital signature service.



   The digital signature service must be exposed to the back-end gateway in transaction */n/IWFND/MAINT_SERVIVCE*. The external service name that must be exposed is *ODATA_DIGITAL_SIGNATURE_SRV*.

   If a front-end service exists, expose the digital signature service in the same way you expose other services.

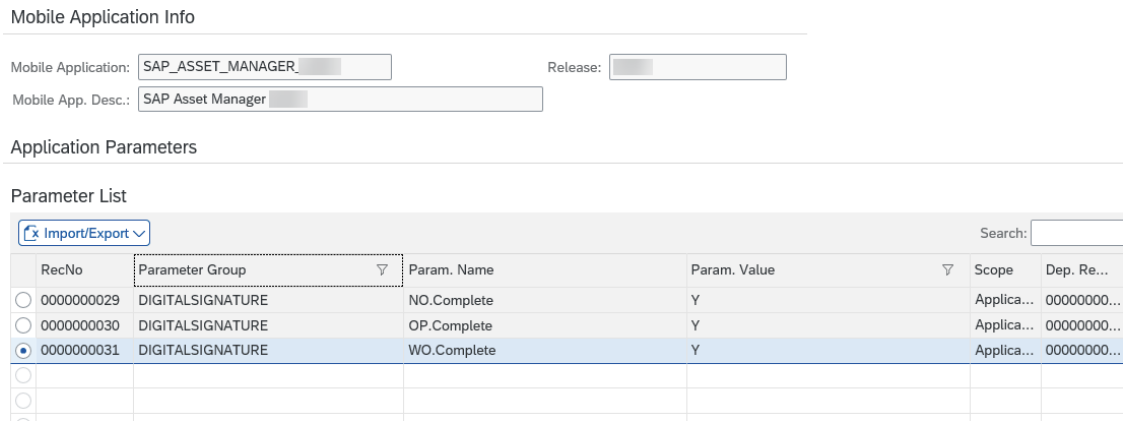
2. **Expose the service on SAP BTP services**: Create a mobile destination and attach it to the SAP Service and Asset Manager 2310 application. The destination name is *DEST_DIGITAL_SIGNATURE_PPROP* with the URL pointing to the back-end service for digital signature.



3. **Enable digital signature on the ConfigPanel:**

   a. Open the ConfigPanel and navigate to *Mobile Application Configuration* > *Parameters tab*. Select your application release.

   b. Enable the *DIGITAL SIGNATURE* parameters required for your site.



   c. *Save* your changes.

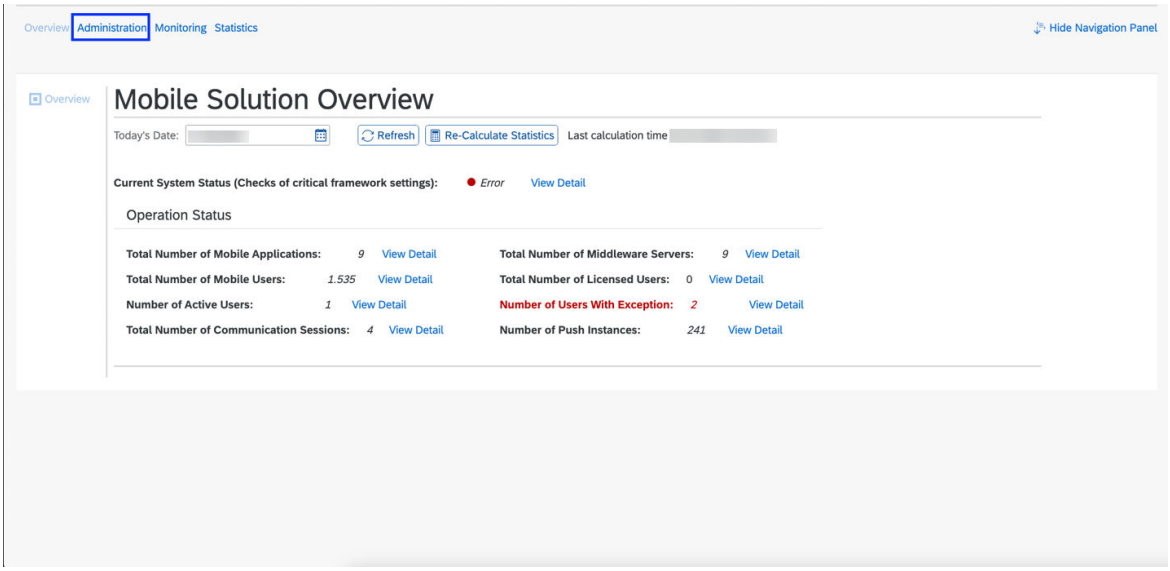
3.2.7.2 Troubleshooting: Unable to Unregister the Device

Procedure

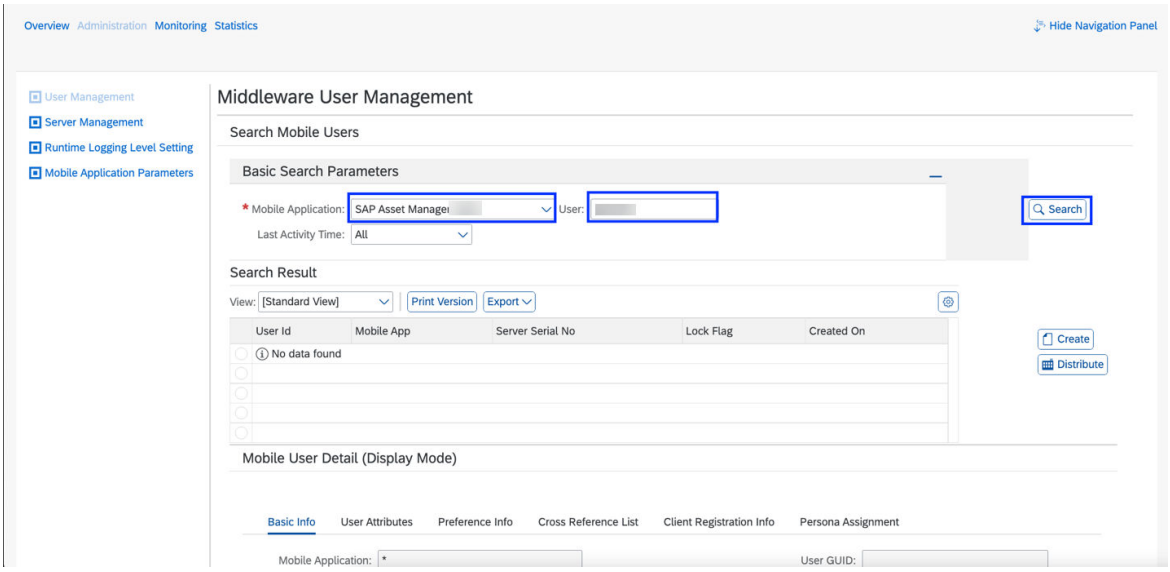
1. Use transaction code *SU3* on the SAP GUI. Choose *TOTP Registration* from the menu options.

   The *Administration of TOTP Devices* window appears. The TOTP Devices table should contain a default entry. If the entry is missing, continue this procedure.

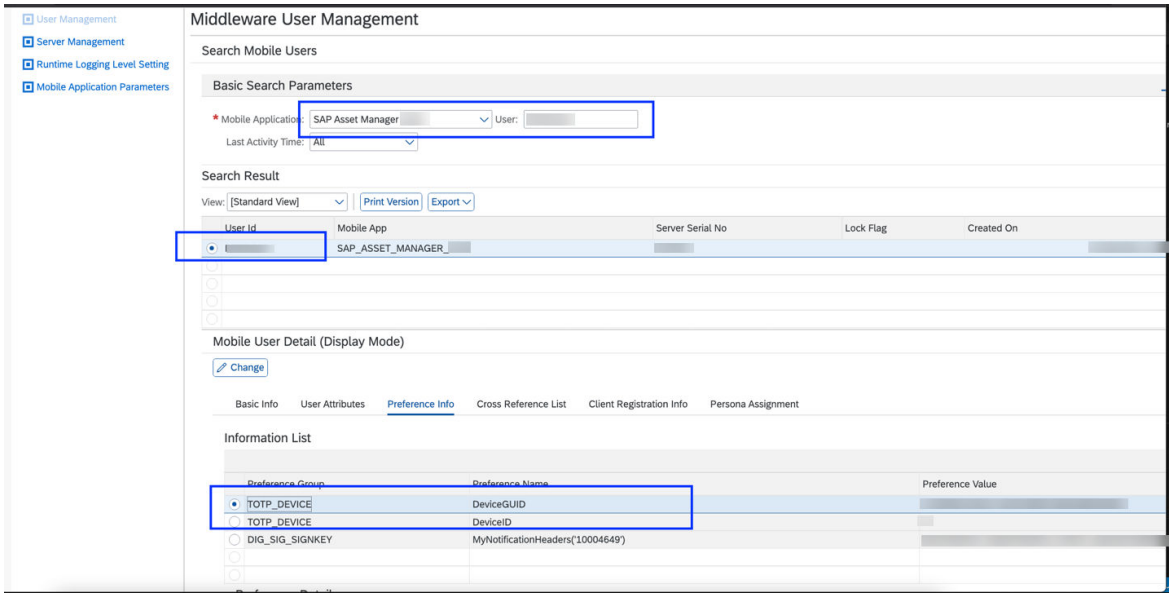
2. Open the Administration & Monitoring Portal through the transaction code */n/SYCLO/ADMIN*. Select the *Administration* tab.



3. Select the *User Management* tab. Provide the `<Mobile Application>` and `<User>` as search input. Click *Search*.



4. Select the appropriate entry from the *Search Result* section. Click the *Preference Info* tab.



5. Delete both *TOTP_DEVICE*-related entries.
6. Open your authenticator app. Delete the old token, then restart the registration process for the device.

3.2.8 Configuring Notification Catalog Types

Use parameters to configure the notification catalog types.

Context

Code groups that belong together in terms of content are grouped in catalogs. These catalogs are identified by the catalog type (a number or a letter). For example, in this way, you combine:

- All code groups for particular problems for a catalog type
- All code for causes for another catalog type and
- All code groups for activities for a further catalog type

Use the *CATALOGTYPE* parameter group and the following parameters within the group to configure your catalog types for notifications in SAP Service and Asset Manager:

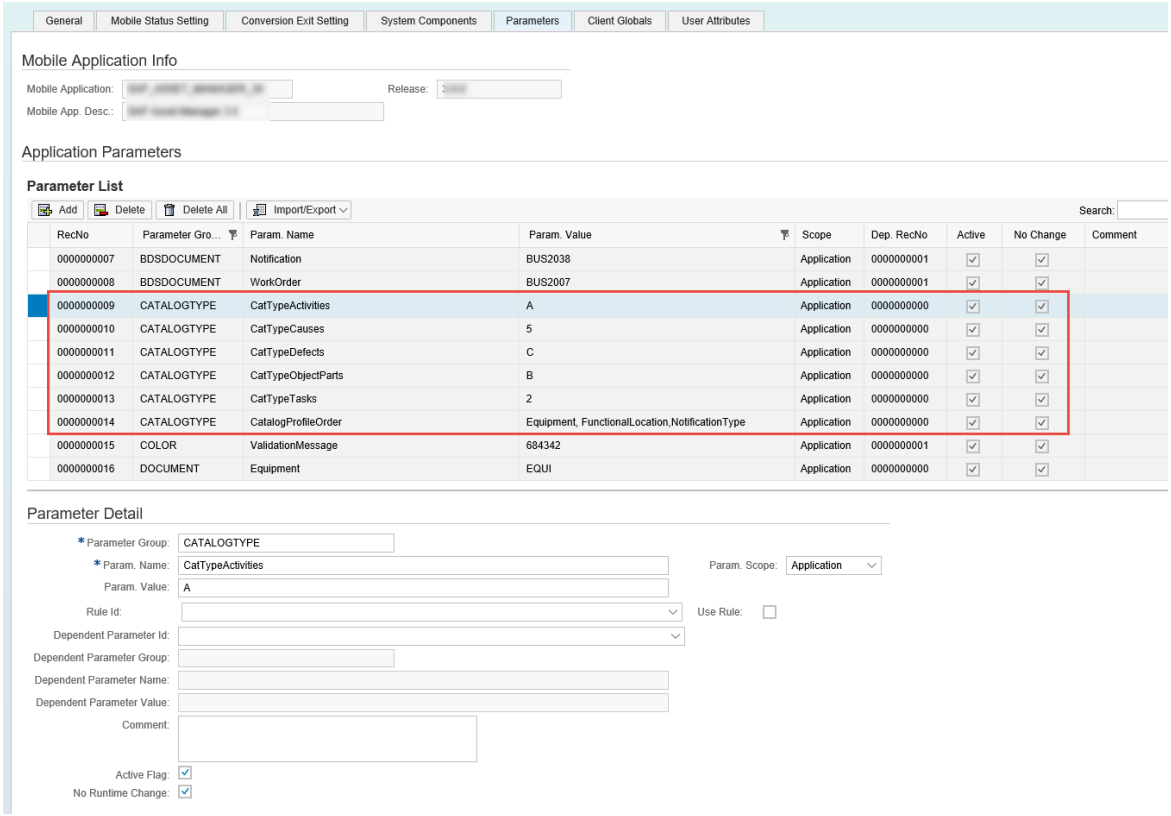
- **CatTypeActivities:** Default is *A*
- **CatTypeCauses:** Default is *5*
- **CatTypeDefects:** Default is *C*
- **CatTypeObjectParts:** Default is *B*
- **CatTypeTasks:** Default is *2*
- **CatalogProfileOrder:** Default is *Equipment, Functional Location, Notification Type*

The *CATALOGTYPE* parameters correspond to the rules found in the OData mobile data object *SAM2310_CATALOG_CODES*. You can add a new data filter rule to your customer namespace, or change the existing parameter-rule association to a new parameter-rule association.

Procedure

1. Using the ConfigPanel, navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.
2. The *CatType[xxx]* parameters are found in the *CATALOGTYPE* group. You can scroll down to find the parameter, or perform a search using the *Search* box. Highlight the parameter you want to configure and click the *Change* button.



3. Make your desired parameter association changes, or change the value of a parameter to *Z*, a custom activity catalog type.
4. Check the `<Active>` flag to ensure that the parameter is used by the mobile application. If desired, and if not already checked, check the `<No Runtime Change>` box to ensure that the value of the parameter is not overridden at runtime through synchronization processing.
5. *Save* your changes.
6. If you are creating a custom activity value type, navigate to *OData Mobile Data Object Configuration* > *Data Filter Tab* > *SAM2310_CATALOG_CODES* > *Operation - READ* > *Standard Filter* > *CATALOG_TYPE*.
7. Click the *Change* button. Add the new value. For information on working with rules, see Working with oData MDO Filter Rules [page 149].
8. *Save* your changes.

3.2.9 Enabling and Disabling Features Per User Through SAP Authorization

Using parameter framework configuration, configure parameters to enable or disable various features per the authorization of the user in the back end.

Each mobile user is connected to a back end SAP user. The back-end SAP user can be assigned one or more roles. These roles grant their holders authorizations within the back end system. Through parameter configuration, SAP provides a standard rule handler that performs a TCode authorization. SAP also provides new globals that can turn on and off new features.

If a parameter is enabled to use a rule instead of a global, and the user role has an authorization to run a specific transaction code, then that specific feature is enabled for that SAP user. If the user has the authorization' for a specific transaction code, then the specific feature is disabled for that mobile user. Therefore, depending on the authorization of the SAP user, the feature now either works or doesn't work, displays, or doesn't display (depending on the feature function), rather than is turned on or off for all users.

Features Available Through SAP Authorization

The following features are available for you to enable or disable. Use the following subsection to learn how to use the ConfigPanel to enable or disable a feature based on the authorization of the user.

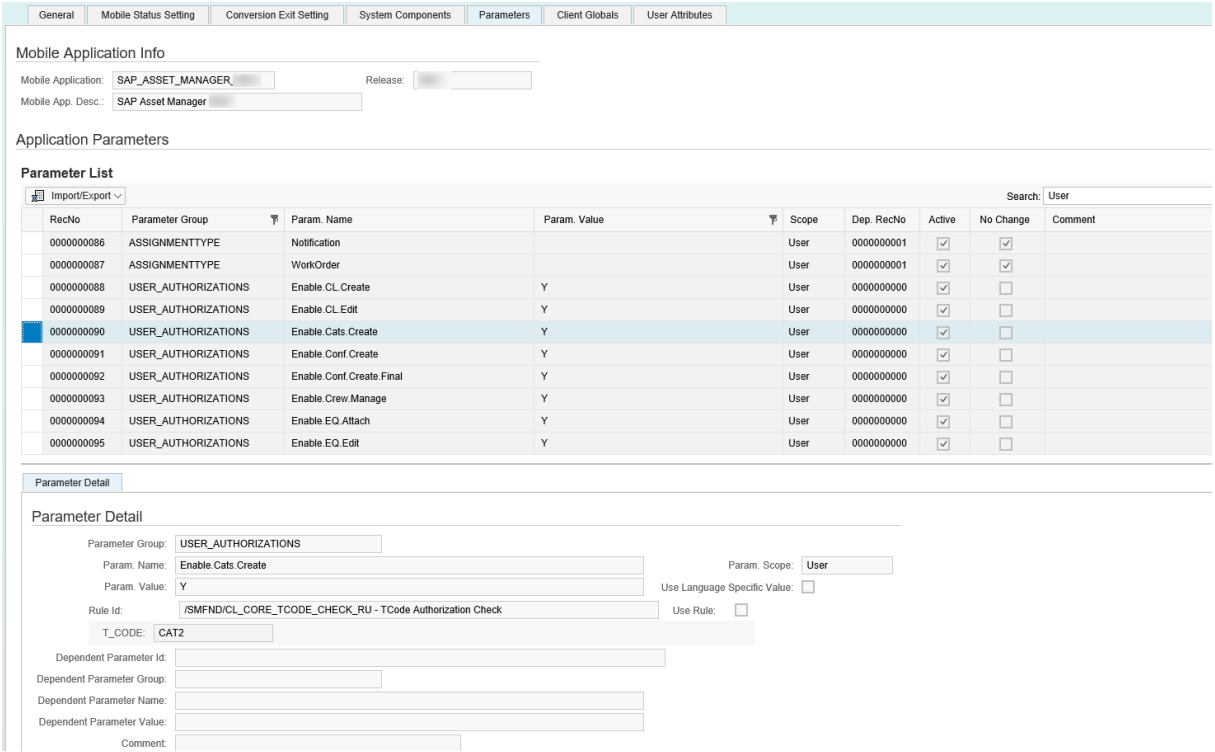
Component	Functionality	Category	TCODE	Back-End Parameter	Comments
SAP ASSET MANAGER	Create work order	Work Orders	IW31	Enable.WO.Create	Includes operations and suboperations
SAP ASSET MANAGER	Edit work order	Work Orders	IW32	Enable.WO.Edit	Includes operations and suboperations (except local)
SAP ASSET MANAGER	Create Notification	Notifications	IW26	Enable.NO.Create	Includes items, tasks, and activities
SAP ASSET MANAGER	Edit Notification	Notifications	IW22	Enable.NO.Edit	Includes items, tasks, and activities (except local)
SAP ASSET MANAGER	Edit FLOC	Functional Location	IL02	Enable.FL.Edit	Includes adding characteristics (except local)
SAP ASSET MANAGER	Edit equip	Equipment	IE02	Enable.EQ.Edit	Includes adding characteristics, install, and dismantle (except local)

Component	Functionality	Category	TCODE	Back-End Parameter	Comments
SAP ASSET MANAGER	Measurement readings	Measurement	IK11	Enable.MD.Create	
SAP ASSET MANAGER	Equip attachment upload	Attachments	N/A	Enable.EQ.Attach	See the *Generic Authorization Check* section in this topic
SAP ASSET MANAGER	FLOC attachment upload	Attachments	N/A	Enable.FL.Attach	See the *Generic Authorization Check* section in this topic
SAP ASSET MANAGER	Allow time recording	CATS	CAT2	Enable.Cats.Create	
SAP ASSET MANAGER	Allow confirmations	Confirmation	IW41	Enable.Conf.Create	
SAP ASSET MANAGER	Allow final confirmation	Confirmation	N/A	Enable.Conf.Create.Final	See the *Generic Authorization Check* section in this topic
SAP ASSET MANAGER	Issue and return parts	MIGO	N/A	Enable.Parts.Issue	See the *Generic Authorization Check* section in this topic
CUSTOMER SERVICE	Service notification create	Notifications	IW51	Enable.SNO.Create	
CUSTOMER SERVICE	Service notification edit	Notifications	IW52	Enable.SNO.Edit	Except local
ASSET CENTRAL	Add checklist	Checklist	N/A	Enable.CL.Create	See the *Generic Authorization Check* section in this topic
ASSET CENTRAL	Fill checklist	Checklist	N/A	Enable.CL.Edit	See the *Generic Authorization Check* section in this topic
CREW	Manage crew	Crew	N/A	Enable.Crew.Manage	See the *Generic Authorization Check* section in this topic
SUPERVISOR	Enable the supervisor role	Supervisor		SupervisorRole	

How to Enable or Disable Features Per User Through Parameters

1. Using the ConfigPanel, navigate from the main screen to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.
   The *Parameter List* table populates with a list of all globals available for the application.
2. Perform a search for the parameter you want to enable or disable as a feature by user role by using the table in this topic to ensure that the parameter is available in the parameter framework for configuration. Search for your parameter in the *Parameter List* using the *Search* box. All user authorization parameters are found under the `<Parameter Group>` name of *USER_AUTHORIZATIONS*. Select your parameter and click the *Change* button.
3. The rule */SMFND/CL_CORE_TCODE_CHECK_RU - TCode Authorization Check* is already selected for you in the `<Rule ID>` field. When you check the `<Use Rule>` checkbox, the rule is active.
4. Change the `<Param. Scope>` dropdown selection from *Application* to *User*.
5. If needed, select the appropriate `<Dependent Parameter ID>` from the dropdown list.
6. Check the `<Active Flag>` checkbox to ensure that your new parameter is active for the user role. *Save* your changes.

See the following screenshot for an example of a configured user role parameter:



Generic Authorization Check

Use the rule */SMFND/CL_CORE_AUTH_CHECK_RU - TCode Authorization Check* to enable a more generic authorization check. This rule is used for the following parameters:

- Enable.EQ.Attach
- Enable.FL.Attach
- Enable.Conf.Create.Final
- Enable.Parts.Issue
- Enable.CL.Create
- Enable.CL.Edit
- Enable.Crew.Manage

To use the */SMFND/CL_CORE_AUTH_CHECK_RU - TCode Authorization Check*, do the following:

1. Create an authorization object. See Authorization Objects for more information.
2. Assign the authorization object to a role.
3. Assign the role to users who will be using the feature.
4. Configure the rule in the ConfigPanel to reference the new authorization object. For more information, see the AUTHORITY-CHECK topic.

3.2.10 Configuring Mobile User Attributes

The User Attributes tab in the Mobile Application Configuration page of the ConfigPanel allows you to maintain multiple values for a selected attribute.

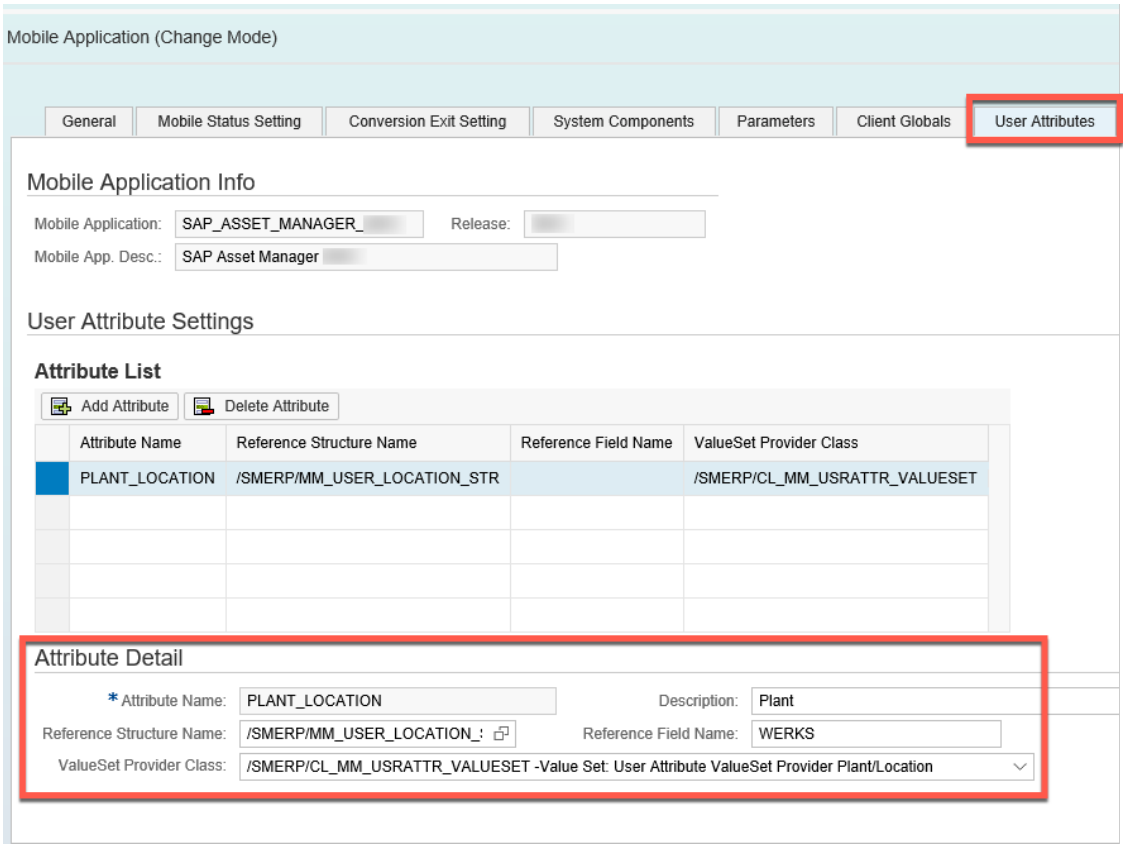
Context

There are three core configuration steps to implement mobile user attributes:

1. Create the user attribute using the ConfigPanel
2. Define and activate the new user attribute through the Admin Portal
3. Define how to consume the data in the OMDO through the ConfigPanel

Procedure

1. **Create the user attribute using the ConfigPanel:**
   a. From the home page of the ConfigPanel, navigate to *Mobile Application Settings* > *Mobile Application Configuration* and select the *User Attributes* tab.
   b. Click the *Change* button. Create your user attribute using the following fields. See the screenshot for an example:



- `<Attribute Name>` and `<Description>`
- In the `<Reference Structure Name>` field, create or use an existing reference structure such as *PLANT* or *WORK CENTER*.
- The `<Reference Field Name>` is optional. Maintain if you're referring to a single field from the structure.
- Create a `<ValueSet Provider Class>` to populate possible values for your new user attribute. Use handler */SMERP/CL_MM_USRATTR_VALUESET* as a reference. Subclass from the base class */SMFND/CL_CORE_USRATTR_VS_BASE*.

c. Click *Save* to save your changes.

2. **Define and activate the new user attribute through the Admin Portal:**

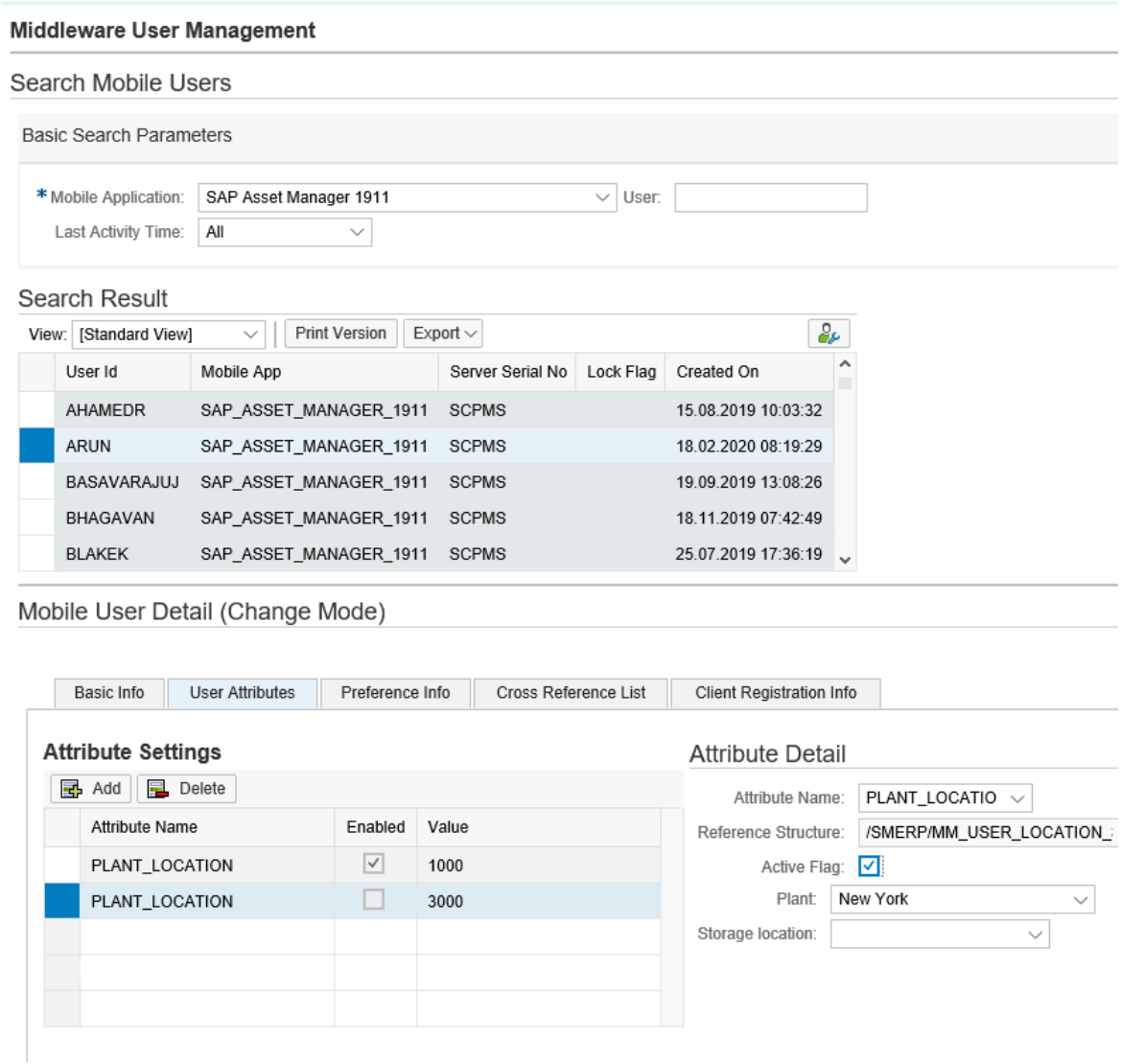
a. Access the Admin Panel from the SAP GUI using transaction code */n/SYCLO/ADMIN*.

b. From the main page of the Admin Portal, navigate to *Administration* > *User Management* and select the *User Attributes* tab in the *Mobile User Detail* section of the screen.

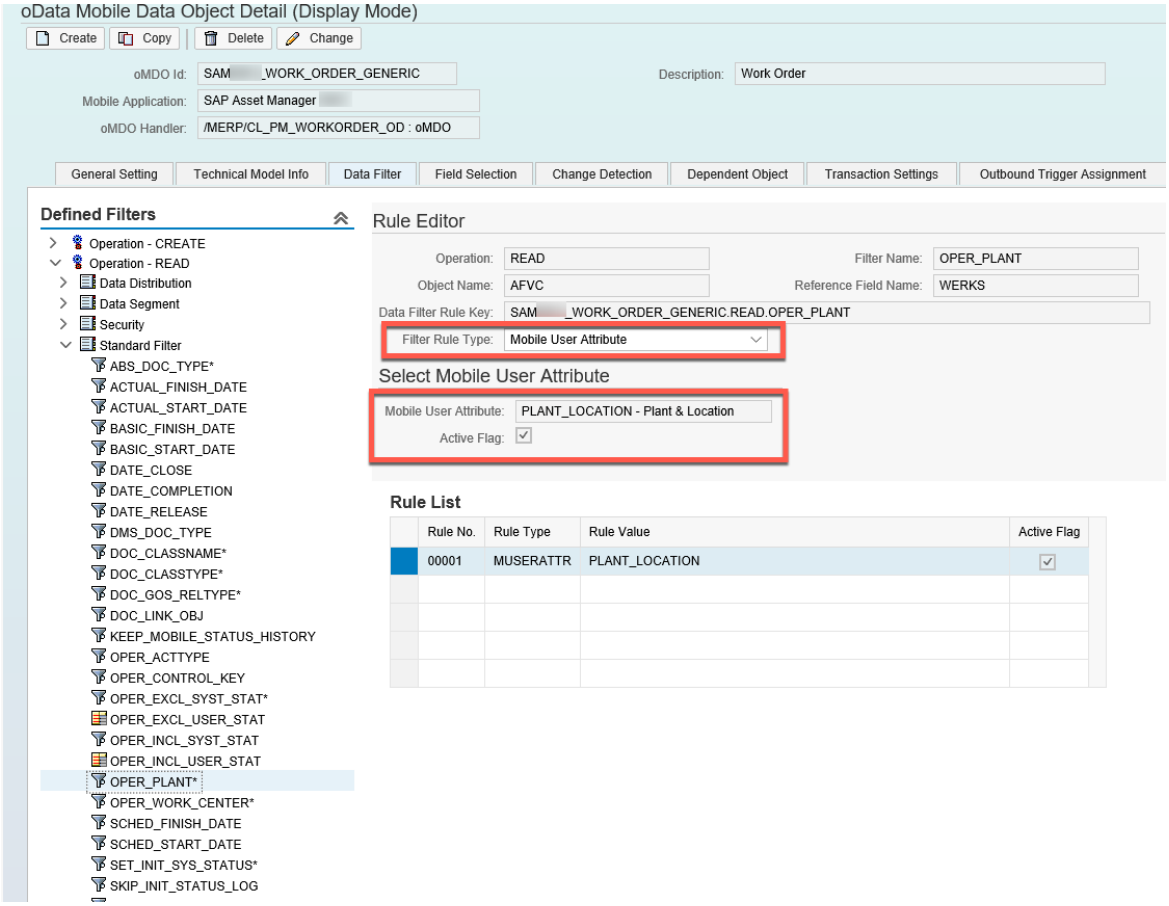
c. Select your desired mobile application from the dropdown. Search for available users connected to your selected application.

d. Click the *User Attributes* tab. Then click the *Change* button.

e. Click the *Add* button. Add the values created in *Step 1*. Repeat this step for all values you created. See the following screenshot for an example.



3. **Define how to consume the data in the OMDO through the ConfigPanel:**
   a. From the main page of the ConfigPanel, navigate to *OData Mobile Data Configuration* > *Data Filter tab*.
   b. Select the OMDO filter to which you're adding the new user attribute from the user attributes defined in the *Mobile Application Configuration* page in the ConfigPanel.
   c. Choose your filter from the *Defined Filters* list. Click *Change*.
   d. Assign your user attribute to the OMDO filter using a dynamic rule. The value is evaluated at runtime based on the runtime mobile user attribute of the user.
   e. Select *Mobile User Attribute* as the `<Filter Rule Type>`. See the following screenshot as an example.



3.2.11 Enabling and Disabling Local Object Mobile Status

The EnableOnLocalBusinessObjects parameter found in the MOBILESTATUS group enables or disables a mobile client user to take a local object through its entire lifecycle even if the client is offline.

Context

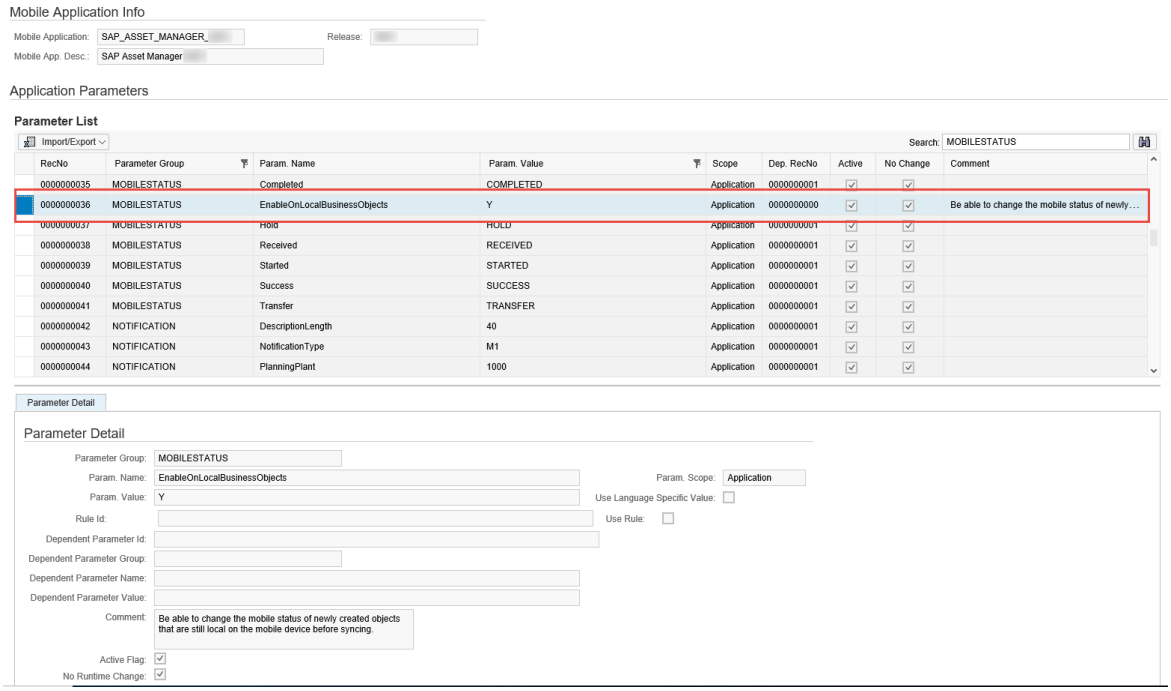
If you enable the *EnableOnLocalBusinessObjects* parameter, SAP Service and Asset Manager allows mobile status update changes for the following:

- Work Order - Assignment level 1
- Operation - Assignment level 2
- Suboperation - Assignment level 3
- Notification
- Notification task
- Notification item task

Procedure

1. Navigate to *Mobile Application Configuration* > *Parameters tab* using the ConfigPanel. Select your application in the left column, *Defined Mobile Applications*.

   The *Parameter List* populates with a list of all parameters available for the application.
2. Find the *EnableOnLocalBusinessObjects* parameter in the *MOBILESTATUS* group. Scroll down to find the parameter, or perform a search using the *Search* box. Highlight the parameter you want to configure and click the *Change* button.



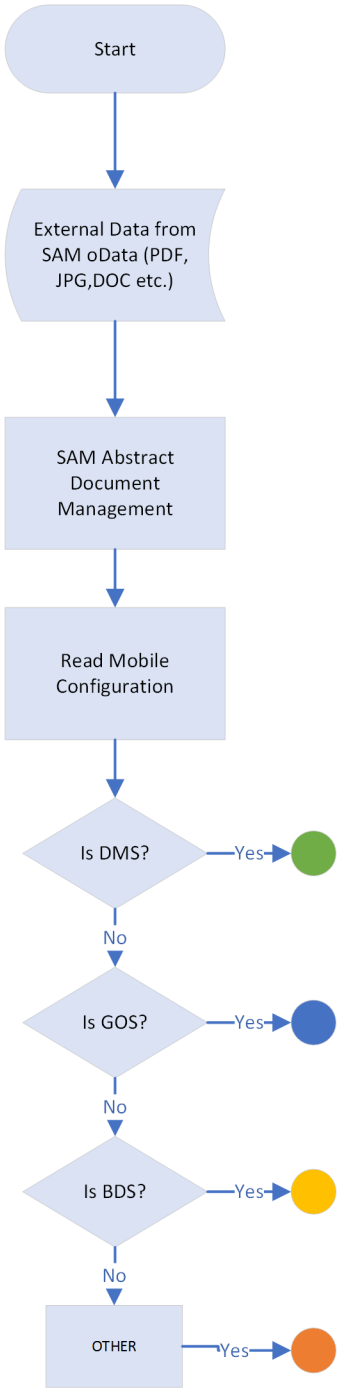
3. Enable or disable the parameter using the following strings: *Y*, *Yes*, *T*, or *True* are used to enable the parameter. *N*, *No*, *F*, or *False* are used to disable the parameter.
4. Check the `<Active>` flag to ensure that the parameter is used by the mobile application. Check the `<No Runtime Change>` box to ensure that the value of the parameter isn't overridden at runtime through synchronization processing if desired and if not already checked.
5. *Save* your changes.

3.2.12 Abstract Document Management

Abstract document management provides the option to create and read DMS, GOS, or BDS documents without configuration at the client application level.

Abstract Document Management Overview

Hover over the different types of abstract document in the flow chart to view the specific flow chart for the document type.



- DMS Method [page 126]
- DMS Method [page 126]
- BDS Method [page 128]
- Other Method [page 129]

3.2.12.1 Configuring Document Management

Abstract document management provides the option to create and read DMS, GOS, or BDS documents without configuration at the client application level.

Configuring Document Management in the ConfigPanel

Prerequisites

Address the following items before performing the procedure:

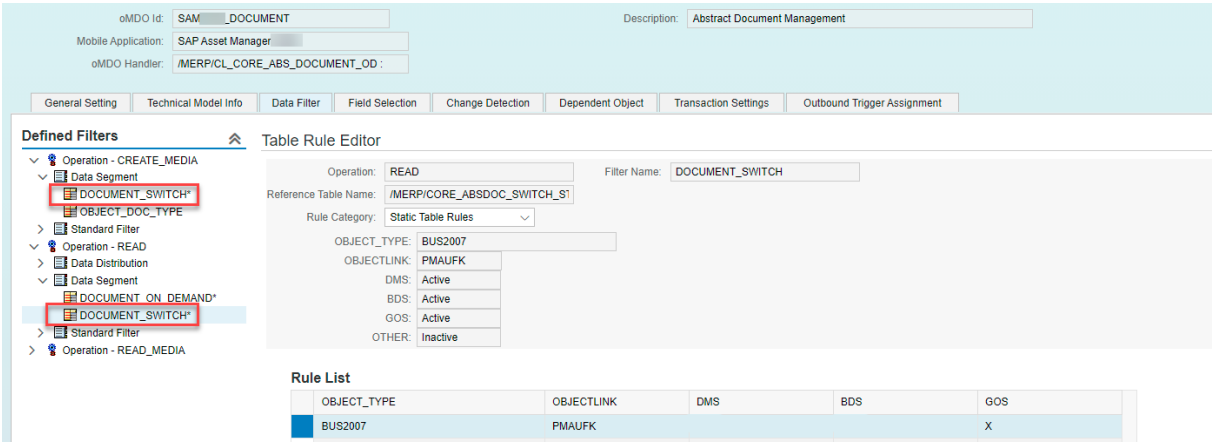
- Know the status or statuses that you're filtering on for equipment synchronization, as they're used in the procedure
- Ability to access to the ConfigPanel and permissions to change configuration settings

With a standard activation of a DMS or GOS document management solution, each document type has its own default content repository. Any application consuming document solutions like DMS or GOS follow the back-end configuration settings.

Select **SAM2310_DOCUMENT** from the *oData Mobile Data Object List* and navigate to the following locations:

- *Operation - CREATE_MEDIA* > *Data Segment* > *DOCUMENT_SWITCH*
- *Operation - READ* > *Data Segment* > *DOCUMENT_SWITCH*

Ensure the repository you're using is set to *Active*:



When you require SAP Service and Asset Manager to create content in a custom repository that is different than your back-end configuration, use the *OTHER* option. Implement the BADI *IF_CORE_OMDO_ABSDOC_BADI* > *CREATE_OTHER_DOCUMENT*. Your implementation must match with the ABS data model to work with standard SAP Service and Asset Manager metadata. See the Other Method [page 129] flow chart for more information.

Configuring the Document Solution to Work with a Third-Party Repository or Content Server

Beginning with the SAP Service and Asset Manager 2010 release, SAP Service and Asset Manager supports working with third-party repositories like Open Text or HTTP. Using third-party repositories leverages the back-end configuration for GOS, BDS, or DMS for work orders, notifications, equipment, and functional locations. See 2945017 to configure older application releases to work with third part repositories.

When using DMS, create custom storage category *ZEXT* to link to a third-party repository, such as an Open Text content server. See 2945017 for more information. Through this additional configuration, attachments uploaded from the application are stored in the third-party content server rather than the default content server.

See the following release notes for addition information on configuring abstract documents:

- 2457912: How to Create a Content Repository in OAC0
- 782614: Using an External Content Server in the BDS
- 530792: Storing Documents in Generic Object Services
- 2571570: Where are Documents Physically Stored with Business Communication Services
- 2945017: SAP Asset Manager ABS Enhancement

> **Note**
>
> It isn't possible to integrate ABS documents with third-party repositories. Use this SAP note to deliver additional parameters such as storage category at an object level.

Document Management Configuration Examples

See the following examples of back-end configuration with a third-party content server:

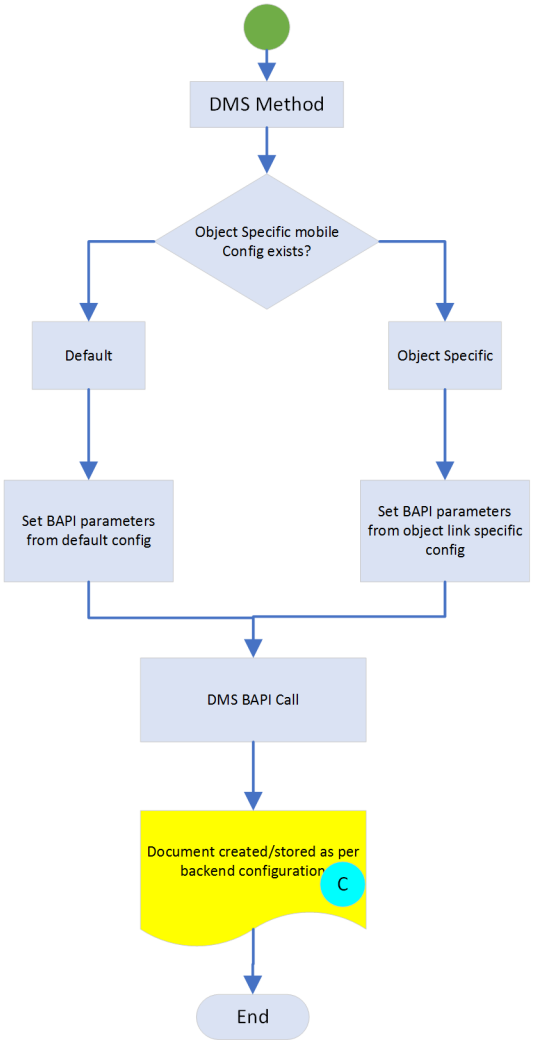
Example 1

Configure third-party repositories like Open Text to work with one of the available document solutions like BDS, DMS, or GOS using the ConfigPanel. Then apply SAP Note 2945017 and configure the desired document solution for work orders, equipment, and functional location objects.

Example 2

Integrate a third-party repository in more than one way with your back-end SAP system. If your implementation doesn't fit with DMS, BDS, or GOS, use the OTHER option in the ConfigPanel. Implement BADI */MERP/CORE_OMDO_ABSDOC_BADI* to align with the implementation.

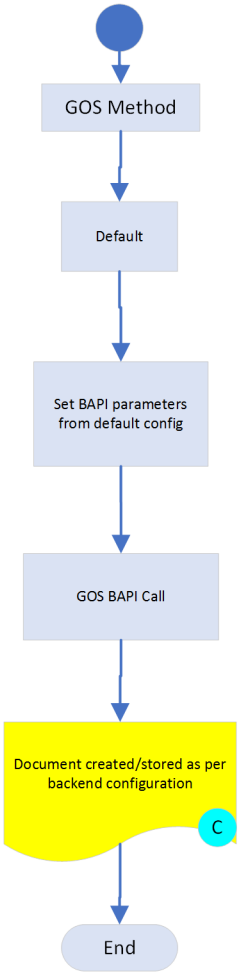
3.2.12.2 DMS Method



Mandatory - SAP backend configuration in working state

C **Optional – Customer specific customisation in backend for DMS, GOS or BDS to store documents in an external repository.**

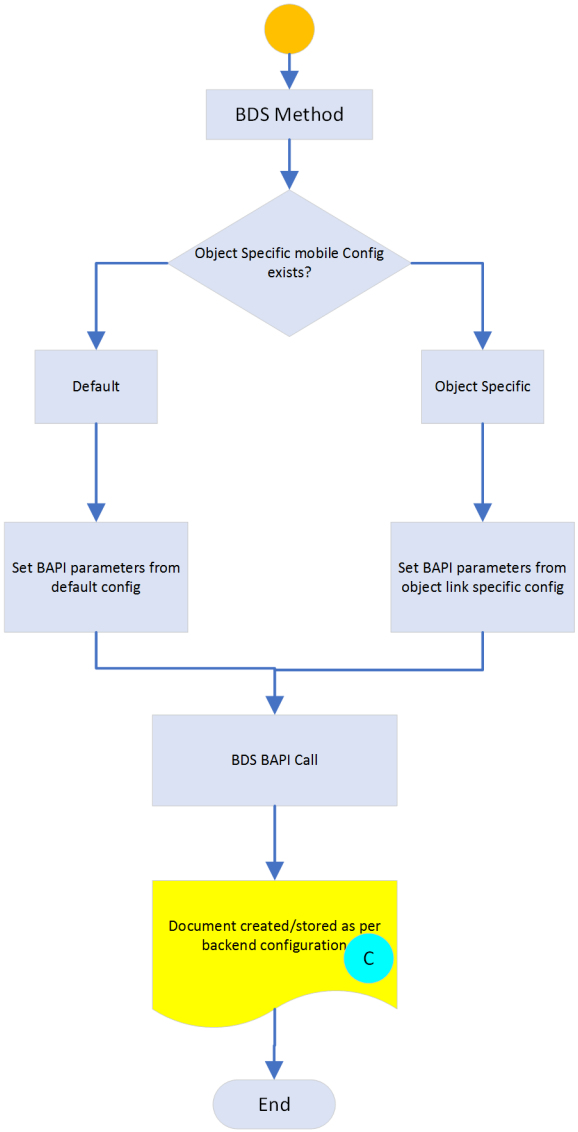
3.2.12.3 GOS Method



Mandatory - SAP backend configuration in working state

C **Optional – Customer specific customisation in backend for DMS, GOS or BDS to store documents in an external repository.**

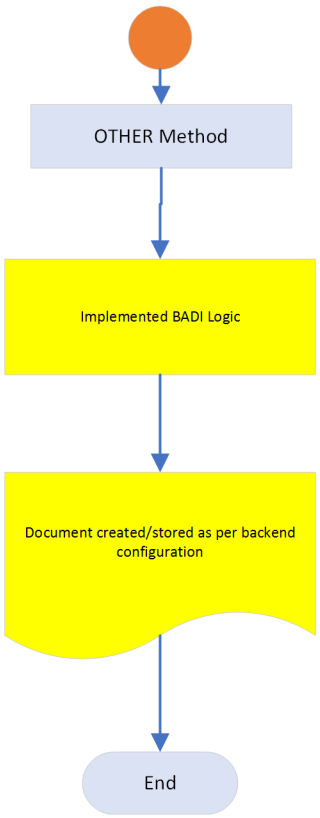
3.2.12.4 BDS Method



Mandatory - SAP backend configuration in working state

C **Optional – Customer specific customisation in backend for DMS, GOS or BDS to store documents in an external repository.**

3.2.12.5 Other Method



Mandatory - SAP backend configuration in working state

3.2.12.6 SAM 2305 Document Data Filter

When taking readings in the SAP Service and Asset Manager mobile app, you can attach documents related to S4 service orders, requests, confirmations, contracts, and related items. Activate S4 on the *SAM2305_DOCUMENT* data filter to attach documents to S4 objects.

1. Select *SAM2305_DOCUMENT* in the *oData Mobile Data Object List* and navigate to the following:
   - *Operation* > *CREATE_MEDIA Data Segment DOCUMENT_SWITCH*
   - *Operation* > *READ Data Segment DOCUMENT_SWITCH*
2. In the Rule List, activate the following objects:
   - BUS2000116: Service Order
   - BUS2000117: Service Confirmation
   - BUS2000223: Service Request

- BUS2000112: Service Contract
- BUS2000140: Service Product Item
- BUS2000146: Service Mat. Item
- BUS2000159: Service Expense Item
- BUS2000142: ServMatConfirmItem
- BUS2000143: ServProdConfirmItem
- BUS2000158: SrvcExpenseConfItem
- BUS2000137: Service Contr. Item

> **Note**
>
> Ensure that both *BDS* and *GOS* are set to *Active*.

3. Navigate to *Operation* > *READ Standard Filter DOC_CLASSNAME*
4. Add a new line to the *Rule List* for each low value and enter the following range value:
   - *Sign*: Inclusive
   - *Option*: =
   - *Low Value*: BUS2000116
5. Activate the following objects:
   - BUS2000117
   - BUS2000223
   - BUS2000112
   - BUS2000140
   - BUS2000146
   - BUS2000159
   - BUS2000142
   - BUS2000143
   - BUS2000158
   - BUS2000137

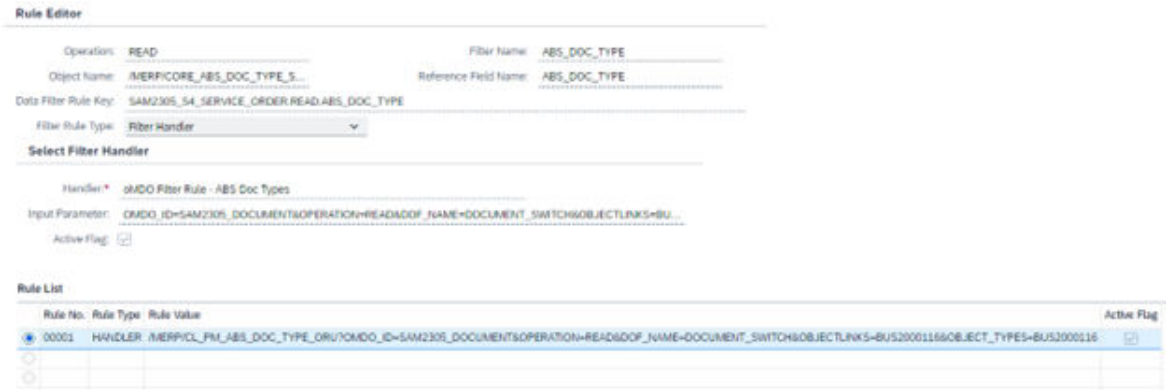
3.2.12.6.1 S4 Service Order

Context

Procedure

1. Select *SAM2305_S4_SERVICE_ORDER* in the *oData Mobile Data Object* list and navigate to the *Data Filter* tab.

2. In the *Defined Filters* list, navigate to *Operation* > *READ Standard Filter ABS_DOC_TYPE*.
3. Set the filter handler *OMDO_ID=SAM2305_DOCUMENT&OPERATION=READ&DOF_NAME=DOCUMENT_SWITCH&OBJECTLINKS=BUS2000116&OBJECT_TYPES=BUS2000116* to *Active*.



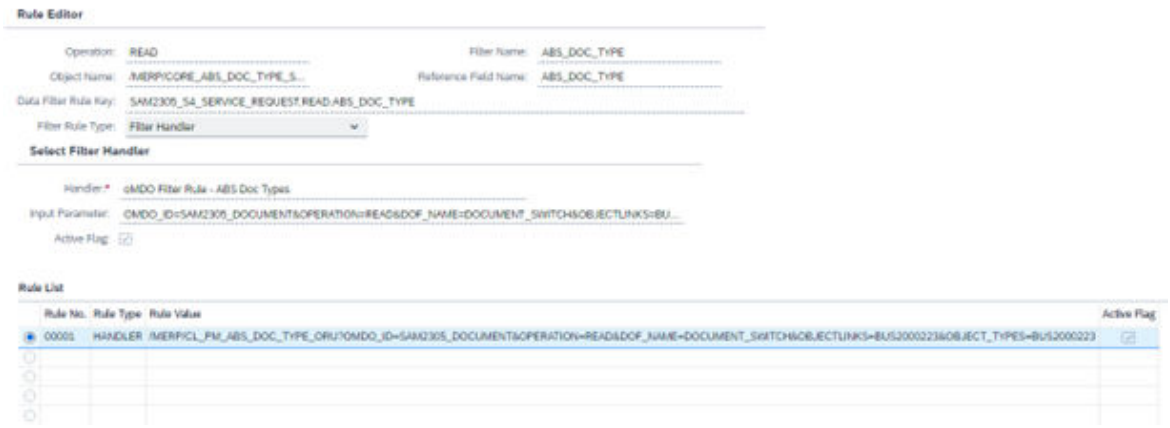
4. Enable dependency between the S4 service order and the related document on the mobile device.
5. Navigate to the *Dependent Object* tab. Find the *S4SERVDOCUMENT* object in the *<Source Tech. Entity Type>* column. Scroll the page to find the *SAM2305_DOCUMENT* object. Ensure it is marked as *Active*.
6. Insert the two lines in the *Dependent Object Keys* tab:
    1. *Source oMDO Field Name:* OBJECT_KEY
       Dependent Object Key Fieldname: OBJECT_KEY
    2. *Source oMDO Field Name:* OBJTYPE_H
       Dependent Object Key Fieldname: OBJECTLINK
7. Insert the two lines in the *Origin Object Keys* tab:
    1. *Field name:* OBJECT_KEY
    2. *Field name:* DOC_OBJ_ID
8. Enable dependency between the S4 service order item and the related document on the mobile device.
9. Navigate to the *Dependent Object* tab. Find *S4SERVITEM* object in the *<Source Tech. Entity Type>* column. Scroll the page to find the *SAM2305_DOCUMENT* object. Ensure it is marked as *Active*.
10. Insert the two lines in the *Dependent Object Keys* tab:
    1. *Source oMDO Field Name:* ITEM_GUID_CHAR
       Dependent Object Key Fieldname: OBJECT_KEY
    2. *Source oMDO Field Name:* OBJTYPE_I
       Dependent Object Key Fieldname: OBJECTLINK
11. Insert the three lines in the *Origin Object Keys* tab:
    1. *Field name:* OBJTYPE_H
    2. *Field name:* OBJECT_ID
    3. *Field name:* NUMBER_INT

3.2.12.6.2 S4 Service Request

Context

Procedure

1. Select *SAM2305_S4_SERVICE_REQUEST* in the *oData Mobile Data Object* list and navigate to the *Data Filter* tab.
2. In the *Defined Filters* list, navigate to *Operation* > *READ Standard Filter ABS_DOC_TYPE*.
3. Set the filter handler *OMDO_ID=SAM2305_DOCUMENT&OPERATION=READ&DOF_NAME=DOCUMENT_SWITCH&OBJECTLINKS=BUS2000223&OBJECT_TYPES=BUS2000223* to *Active*.



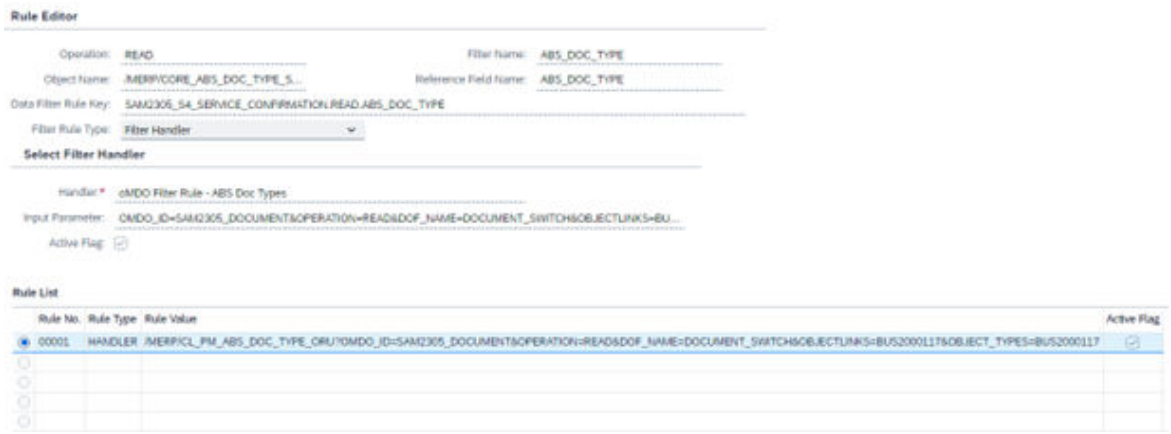
4. Enable dependency between the S4 service request and the related document on the mobile device.
5. Navigate to the *Dependent Object* tab. Find the *S4SERVDOCUMENT* object in the *<Source Tech. Entity Type>* column. Scroll the page to find the *SAM2305_DOCUMENT* object. Ensure it is marked as *Active*.
6. Insert the two lines in the *Dependent Object Keys* tab:
    1. *Source oMDO Field Name:* OBJECT_KEY
       Dependent Object Key Fieldname: OBJECT_KEY
    2. *Source oMDO Field Name:* OBJTYPE_H
       Dependent Object Key Fieldname: OBJECTLINK
7. Insert the two lines in the *Origin Object Keys* tab:
    1. *Field name:* OBJECT_KEY
    2. *Field name:* DOC_OBJ_ID

3.2.12.6.3 S4 Service Confirmation

Context

Procedure

1. Select *SAM2305_S4_SERVICE_CONFIRMATION* in the *oData Mobile Data Object* list and navigate to the *Data Filter* tab.
2. In the *Defined Filters* list, navigate to *Operation* > *READ Standard Filter ABS_DOC_TYPE*.
3. Set the filter handler *OMDO_ID=SAM2305_DOCUMENT&OPERATION=READ&DOF_NAME=DOCUMENT_SWITCH&OBJECTLINKS=BUS2000117&OBJECT_TYPES=BUS2000117* to *Active*.



4. Enable dependency between the S4 service confirmation and the related document on the mobile device.
5. Navigate to the *Dependent Object* tab. Find the *S4SERVDOCUMENT* object in the *<Source Tech. Entity Type>* column. Scroll the page to find the *SAM2305_DOCUMENT* object. Ensure it is marked as *Active*.
6. Insert the two lines in the *Dependent Object Keys* tab:
    1. *Source oMDO Field Name:* OBJECT_KEY
       Dependent Object Key Fieldname: OBJECT_KEY
    2. *Source oMDO Field Name:* OBJTYPE_H
       Dependent Object Key Fieldname: OBJECTLINK
7. Insert the two lines in the *Origin Object Keys* tab:
    1. *Field name:* OBJECT_KEY
    2. *Field name:* DOC_OBJ_ID
8. Enable dependency between the S4 service confirmation item and the related document on the mobile device.
9. Navigate to the *Dependent Object* tab. Find *S4SERVITEM* object in the *<Source Tech. Entity Type>* column. Scroll the page to find the *SAM2305_DOCUMENT* object. Ensure it is marked as *Active*.

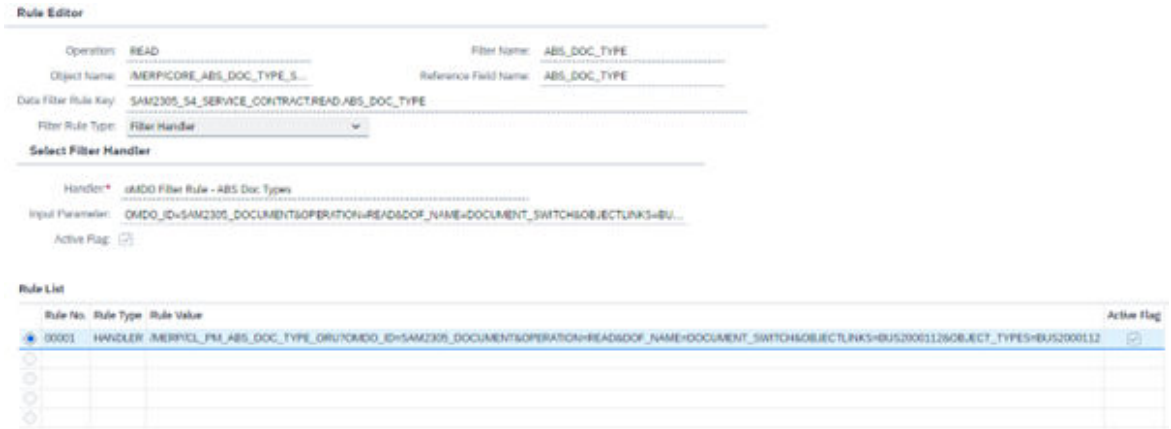
10. Insert the two lines in the *Dependent Object Keys* tab:
    1. *Source oMDO Field Name:* ITEM_GUID_CHAR
       Dependent Object Key Fieldname: OBJECT_KEY
    2. *Source oMDO Field Name:* OBJTYPE_I
       Dependent Object Key Fieldname: OBJECTLINK
11. Insert the three lines in the *Origin Object Keys* tab:
    1. *Field name:* OBJTYPE_H
    2. *Field name:* OBJECT_ID
    3. *Field name:* NUMBER_INT

3.2.12.6.4 S4 Service Contract

Context

Procedure

1. Select *SAM2305_S4_SERVICE_CONTRACT* in the *oData Mobile Data Object* list and navigate to the *Data Filter* tab.
2. In the *Defined Filters* list, navigate to *Operation* > *READ Standard Filter ABS_DOC_TYPE*.
3. Set the filter handler *OMDO_ID=SAM2305_DOCUMENT&OPERATION=READ&DOF_NAME=DOCUMENT_SWITCH&OBJECTLINKS=BUS2000112&OBJECT_TYPES=BUS2000112* to *Active*.



4. Enable dependency between the S4 service contract and the related document on the mobile device.
5. Navigate to the *Dependent Object* tab. Find the *S4SERVDOCUMENT* object in the *<Source Tech. Entity Type>* column. Scroll the page to find the *SAM2305_DOCUMENT* object. Ensure it is marked as *Active*.

6. Insert the two lines in the *Dependent Object Keys* tab:
    1. *Source oMDO Field Name:* OBJECT_KEY
       Dependent Object Key Fieldname: OBJECT_KEY
    2. *Source oMDO Field Name:* OBJTYPE_H
       Dependent Object Key Fieldname: OBJECTLINK
7. Insert the two lines in the *Origin Object Keys* tab:
    1. *Field name:* OBJECT_KEY
    2. *Field name:* DOC_OBJ_ID
8. Enable dependency between the S4 service contract item and the related document on the mobile device.
9. Navigate to the *Dependent Object* tab. Find *S4SERVITEM* object in the *<Source Tech. Entity Type>* column. Scroll the page to find the *SAM2305_DOCUMENT* object. Ensure it is marked as *Active*.
10. Insert the two lines in the *Dependent Object Keys* tab:
    1. *Source oMDO Field Name:* ITEM_GUID_CHAR
       Dependent Object Key Fieldname: OBJECT_KEY
    2. *Source oMDO Field Name:* OBJTYPE_I
       Dependent Object Key Fieldname: OBJECTLINK
11. Insert the three lines in the *Origin Object Keys* tab:
    1. *Field name:* OBJTYPE_H
    2. *Field name:* OBJECT_ID
    3. *Field name:* NUMBER_INT

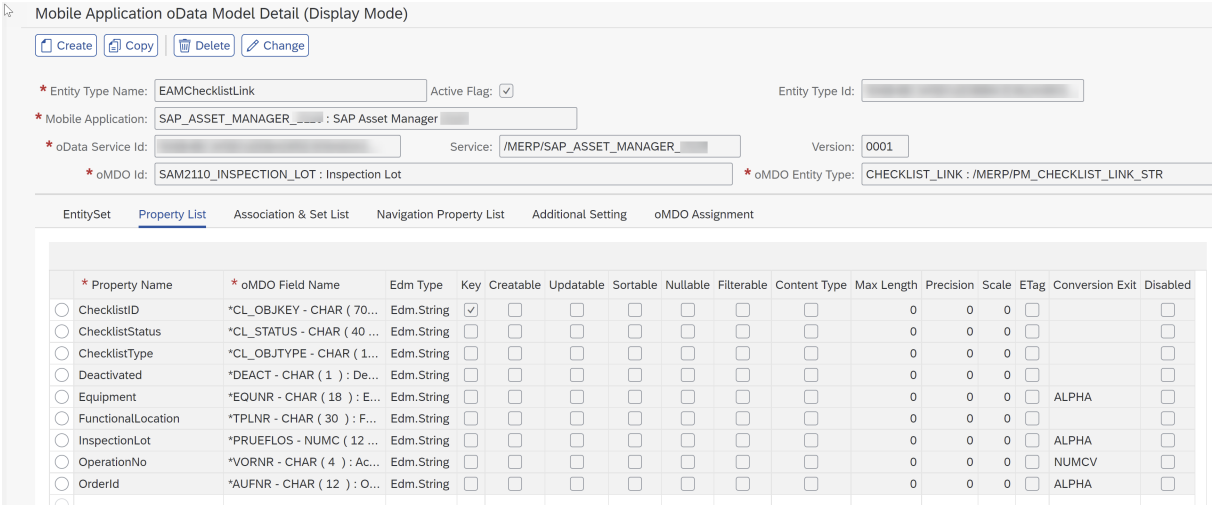
3.2.13 EAM Checklist Overview

EAM processes need checklists for actions such as inspections and maintenance. SAP Service and Asset Manager uses EAM checklists with work orders and operations.

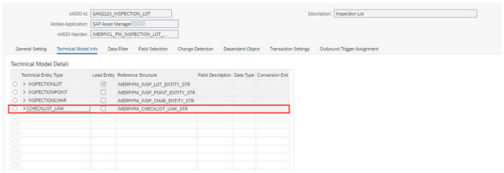
> **Note**
>
> This core feature is only available on an SAP S/4HANA 2021 or later back end system. This feature is therefore available for SAP Service and Asset Manager 2110 or later releases installed on an SAP S/4HANA or later back end system.

Find the EAM checklist feature in the OMDO handler */MERP/CL_PM_INSPECTION_LOT_OD*, as an existing entity. Features of inspection lot are reused to implement the EAM checklist functionality, along with a new entity, *EAMChecklistLink*. Access the OMDO handler through transaction *SE24* on your back end system.

You can view the new entity in the ConfigPanel by navigating to *oData Model Configuration* > *Mobile App SAP_ASSET_MANAGER_<version>* > */MERP/SAP_ASSET_MANAGER_<version>* > *EAMChecklistLink*.



The *INSPECTION_LOT* OMDO is updated to add a technical entry for checklists. Navigate to *oData Mobile Data Object Configuration* > *Technical Model Info tab* > *<version>_INSPECTION_LOT*. Be sure to choose your application release version using the *Mobile Application Filter*.



Use the following procedures to configure the EAM checklist feature:

- Enabling or Disabling the EAM Checklist [page 136]
- Enabling or Disabling Manual Defect Recording [page 137]

3.2.13.1 Enabling or Disabling the EAM Checklist

Context

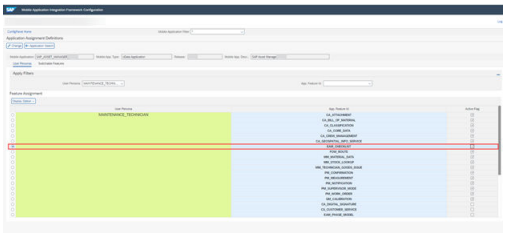
The EAM Checklist feature is called *EAM_CHECKLIST*. By default, the feature is not enabled.

> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

Procedure

1. Navigate to *Component Assignments*. Select your application in the *Mobile Application Filter*. Click the application link in the *Search Result* table.

   The *Application Assignment Definitions* page displays.
2. Click the *Change* button. Find the *EAM_CHECKLIST* feature ID in the *MAINTENANCE_TECHNICIAN* user persona part of the table. Enable or disable the feature using the *Active Flag* checkbox.



3. *Save* your changes.

   If you've enabled the checklist feature, the inspection results display on a mobile device on the *Checklists* detail screen. Validation of all inspections results is done per limits set during configuration of inspection characteristics in the inspection plan. Once results on the mobile device are synced to the back end, the results and usage decision are updated. The EAM checklist is marked as complete on the mobile device.
4. Enable the dependency between a work order and a related inspection lot on the mobile device. Navigate to *SAM2205_WORK_ORDER_GENERIC oMDO*.
5. Select the *Dependent Object* tab. Find the *WOHEADER* objects in the *<Source Tech. Entity Type>* column.
6. Find the *SAM<version>_INSPECTION_LOT* object. Enable the feature using the *Active* checkbox.

3.2.13.2 Enabling or Disabling Manual Defect Recording

Context

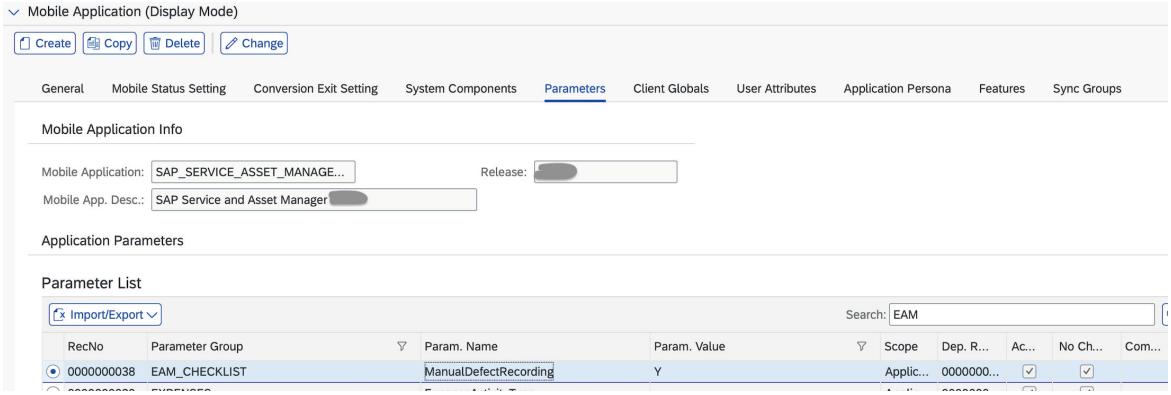
The functionality for creating defects manually for a rejected inspection task result from the mobile device is enabled through the *Parameters* tab. The EAM checklist feature is enabled by default.

Procedure

1. Using the ConfigPanel, navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.

2. The EAM checklist parameters are found in the *EAM_CHECKLIST* group. You can scroll down to find the parameter, or perform a search using the *Search* box. Highlight the *ManualDefectRecording* parameter and click the *Change* button.



The parameters found in the *EAM_CHECKLIST* group work alongside the order type configuration to automatically create the defects upon results recording. When the parameter is enabled, the *Record Defct* button on a rejected inspection characteristic is availabe on the app. A technician can add defect details. The app creates a notification on the back end when the mobile device is synced for the default EAM defect notification type assigned to that order type.

3. Make your desired parameter association changes. Enable the parameter as follows:

   - **Y:** Select if work order types are not set up with automatic defect recording
   - **N:** Select if you've configured work order types set up with automatic defect recording

4. Check the `<Active>` flag to ensure that the parameter is used by the mobile application. If desired, and if not already checked, check the `<No Runtime Change>` box to ensure that the value of the parameter is not overridden at runtime through synchronization processing.
5. *Save* your changes.

3.2.13.3 EAM Checklist Attachments

When enabling EAM on the SAM2305_DOCUMENT data filter, you can attach documents to inspection lots. And when taking readings in the SAP Service and Asset Manager mobile app, you can attach documents related to the inspection lots.

1. Select *SAM2305_DOCUMENT* in the *oData Mobile Data Object List* and navigate to the following:
   - *Operation* > *CREATE_MEDIA Data Segment DOCUMENT_SWITCH*
   - *Operation* > *READ Data Segment DOCUMENT_SWITCH*
2. Add new lines to the *Rule List* for *BDS* and *GOS*:
   - *OBJECT_TYPE*: BUS2045
   - *OBJECTLINK*: BUS2045

   > **Note**
   >
   > Ensure that both *BDS* and *GOS* are set to *Active*.

3. Add new lines to the *Rule List* for *DMS*:

- *OBJECT_TYPE*: BUS2045
- *OBJECTLINK*: QALS

> **Note**
>
> Ensure that both *DMS* is set to *Active*.

4. Navigate to *Operation* > *READ Standard Filter DOC_LINK_OBJ*
5. Add a new line to the *Rule List* and enter the following range value:
   - *Sign*: Inclusive
   - *Option*: =
   - *Low Value*: QALS
6. Enable the active flag
7. Navigate to *Operation* > *READ Standard Filter CLASSNAME*
8. Add a new line to the *Rule List* and enter the following range value:
   - *Sign*: Inclusive
   - *Option*: =
   - *Low Value*: BUS2045
9. Enable the active flag.

3.2.13.3.1 SAM 2305 Inspection Lot

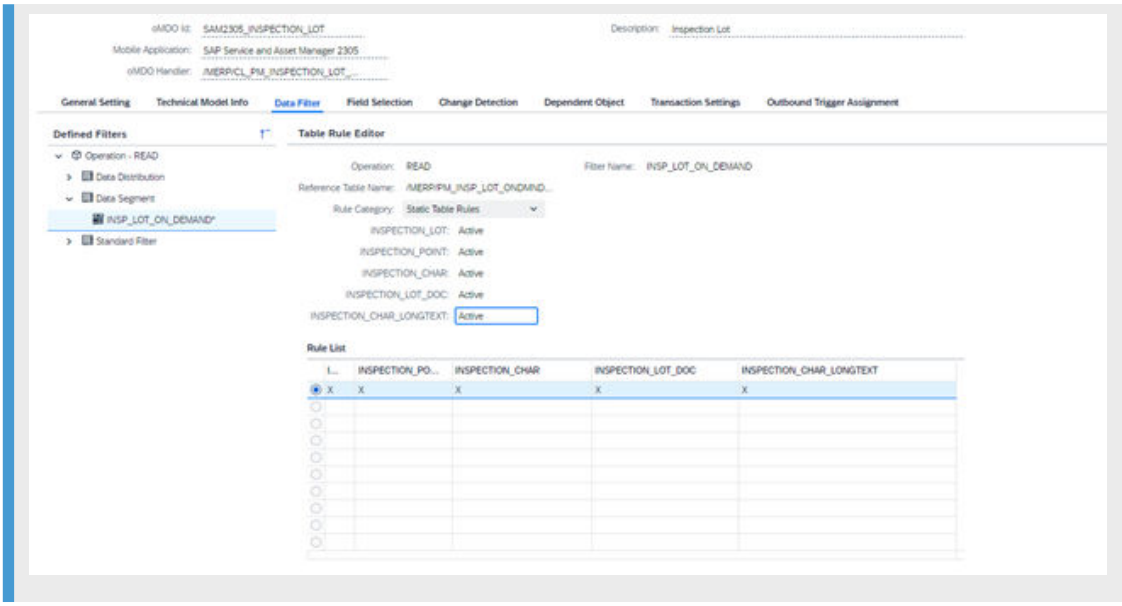
Context

Procedure

1. Select *SAM2305_INSPECTION_LOT* in the *oData Mobile Data Object List* and navigate to the *Data Filter* tab.
2. In the *Defined Filters* list, navigate to *Operation* > *READ Standard Filter ABS_DOC_TYPE*
3. Set filter handler *OMDO_ID=SAM2305_DOCUMENT&OPERATION=READ&DOF_NAME=DOCUMENT_SWITCH&OBJECTLINKS=BUS2045,QALS&OBJECT_TYPES=BUS2045* to *Active*.

4. Navigate to *Operation* > *READ Standard Filter DOC_CLASSNAME*
5. Add a new line to the *Rule List* and enter the following range value:
    - *Sign*: Inclusive
    - *Option*: =
    - *Low Value*: BUS2045
6. Enable the active flag
7. Navigate to *Operation* > *READ Standard Filter DOC_CLASSTYPE*
8. Add a new line to the *Rule List* and enter the following range value:
    - *Sign*: Inclusive
    - *Option*: =
    - *Low Value*: *BO* > *Business Object Repository* object
9. Enable the active flag
10. Navigate to *Operation* > *READ Standard Filter DOC_GOS_RELTYPE*
11. Add a new line to the *Rule List* and enter the following range value:
    - *Sign*: Inclusive
    - *Option*: =
    - *Low Value*: *ATTA* > *Has the Attachment*
12. Enable the active flag
13. Add a new line to the *Rule List* and enter the following range value:
    - *Sign*: Inclusive
    - *Option*: =
    - *Low Value*: *URL* > *Has Internet Link*
14. Enable the active flag
15. Enable dependency between the inspection lot and the related document on the mobile device
16. Navigate to the *Dependent Object* tab. Find *INSPECTIONLOTDOC* objects in the *<Source Tech. Entity Type>* column. Scroll the page to find the *SAM2305_DOCUMENT* object. Ensure it is marked as *Active*.
17. Insert new lines in the *Dependent Object Keys* tab:
    1. *Source oMDO Field Name:* OBJECT_KEY
       Dependent Object Key Fieldname: OBJECT_KEY
18. Insert new lines in the *Origin Object Keys* tab:
    1. *Field name:* OBJECT_KEY
    2. *Field name:* DOC_OBJ_ID

> ### Note
>
> As of 2305 version, the SAP Service and Asset Manager introduces an additional feature concerning long texts for master inspection characteristics: a new *MasterInspectionCharLongText* entity is created as an *InspectionLot* sub-entity. Set the flag to *Insp_Lot_On_Demand* to enable reading the long text of the main inspection characteristic. By default, it is active.



3.2.14 Configuring the Creation of Technical Objects from a Template

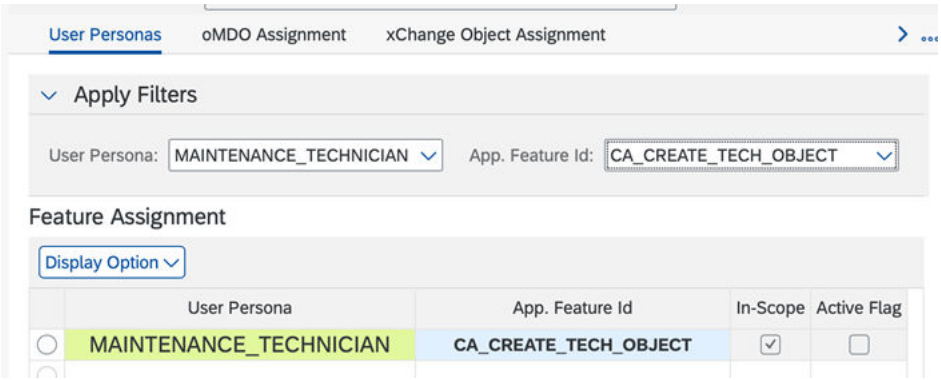
You can create Equipment and Functional Locations from the SAP Service and Asset Manager application in a similar way to executing transactions IE01 and IL01 from the SAP Service and Asset Manager GUI.

Context

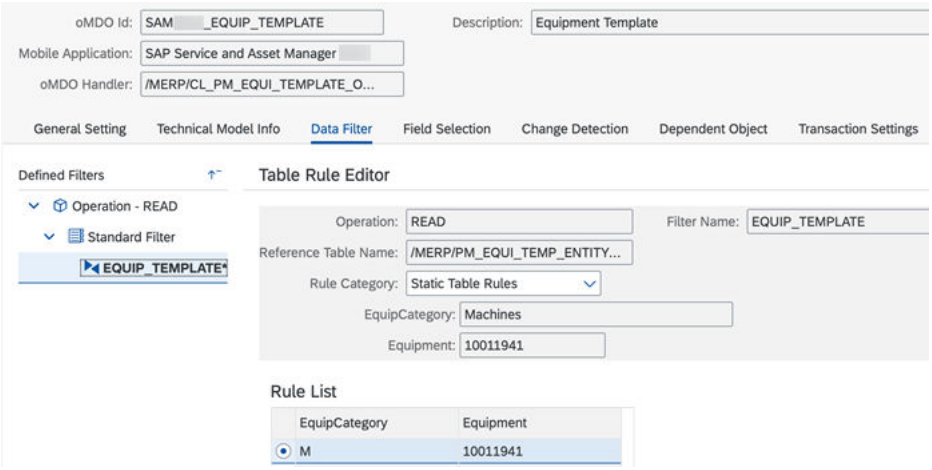
A template is required when you create equipment, but it's optional when you create a functional location. Any existing equipment or functional location can be used as a template. It's possible to copy the classifications, measuring points, business partners, documents, install location (equipment only), and notes from the template. The description, maintenance plant, start date, manufacturer, date of manufacturer, model number, serial number, and room are populated from the template object and you can overwrite them. It's possible to update equipment and functional location before syncing.

Procedure

1. To access the ConfigPanel using a direct transaction code shortcut, enter `/n/syclo/configpanel`.
2. Activate CA_CREATE_TECH_OBJECT for MAINTENANCE_TECHNICIAN Persona in the *Component Assignments*.

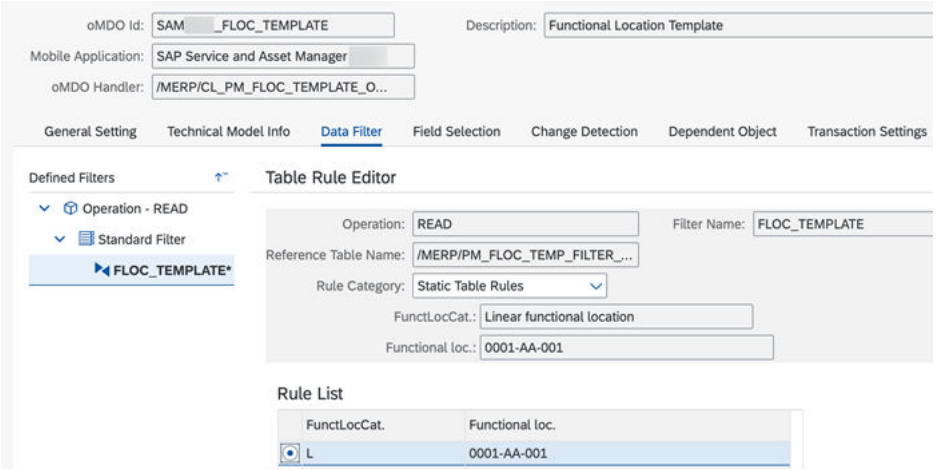


3. Add a rule to EQUIP_TEMPLATE filter in the SAM<version>_EQUIP_TEMPLATE oMDO in the oData Mobile Data Object Configuration.



The following step is optional:

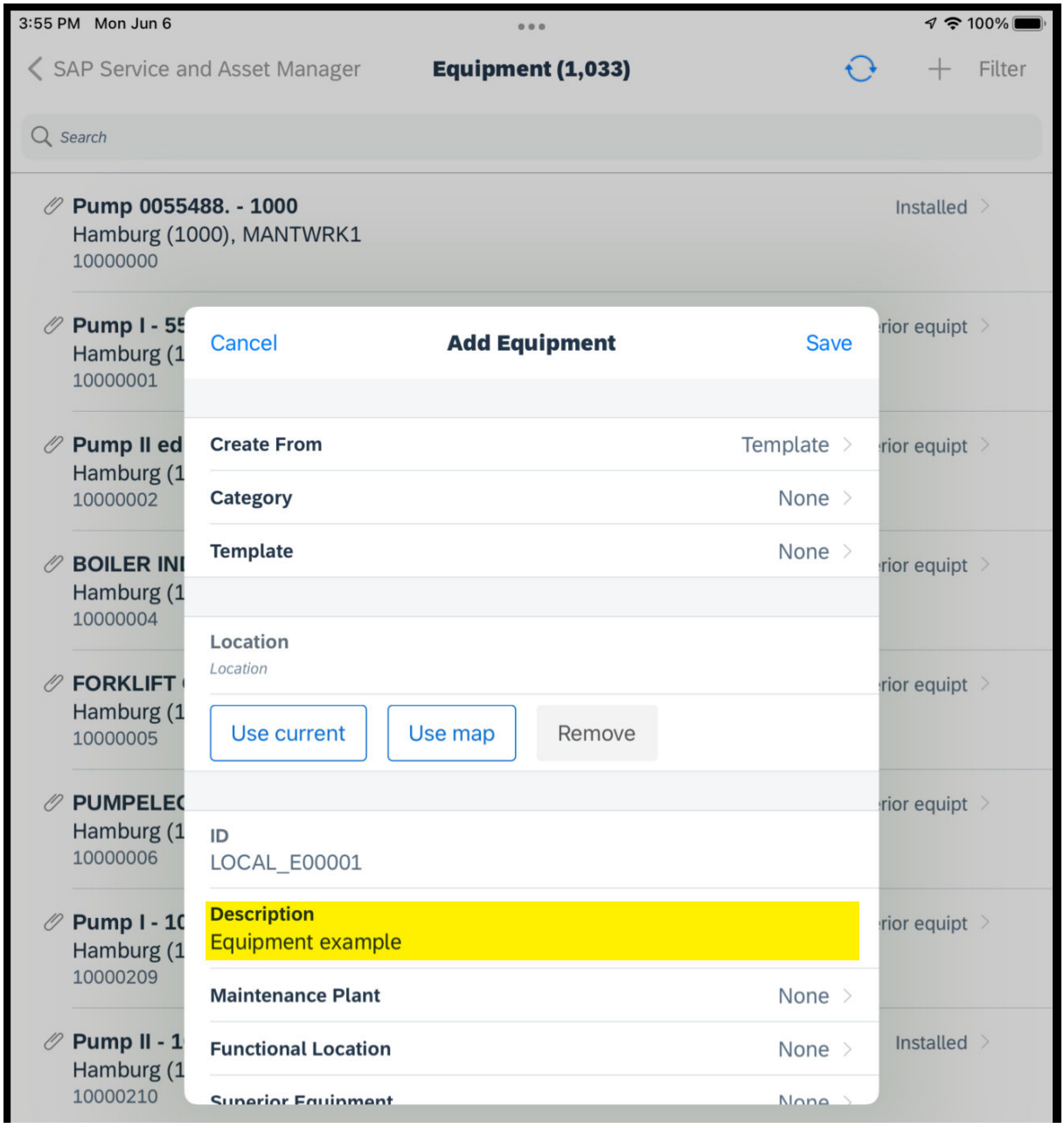
4. Add a rule to FLOC_TEMPLATE filter in SAM<version>_FLOC_TEMPLATE oMDO in the oData Mobile Data Object Configuration.

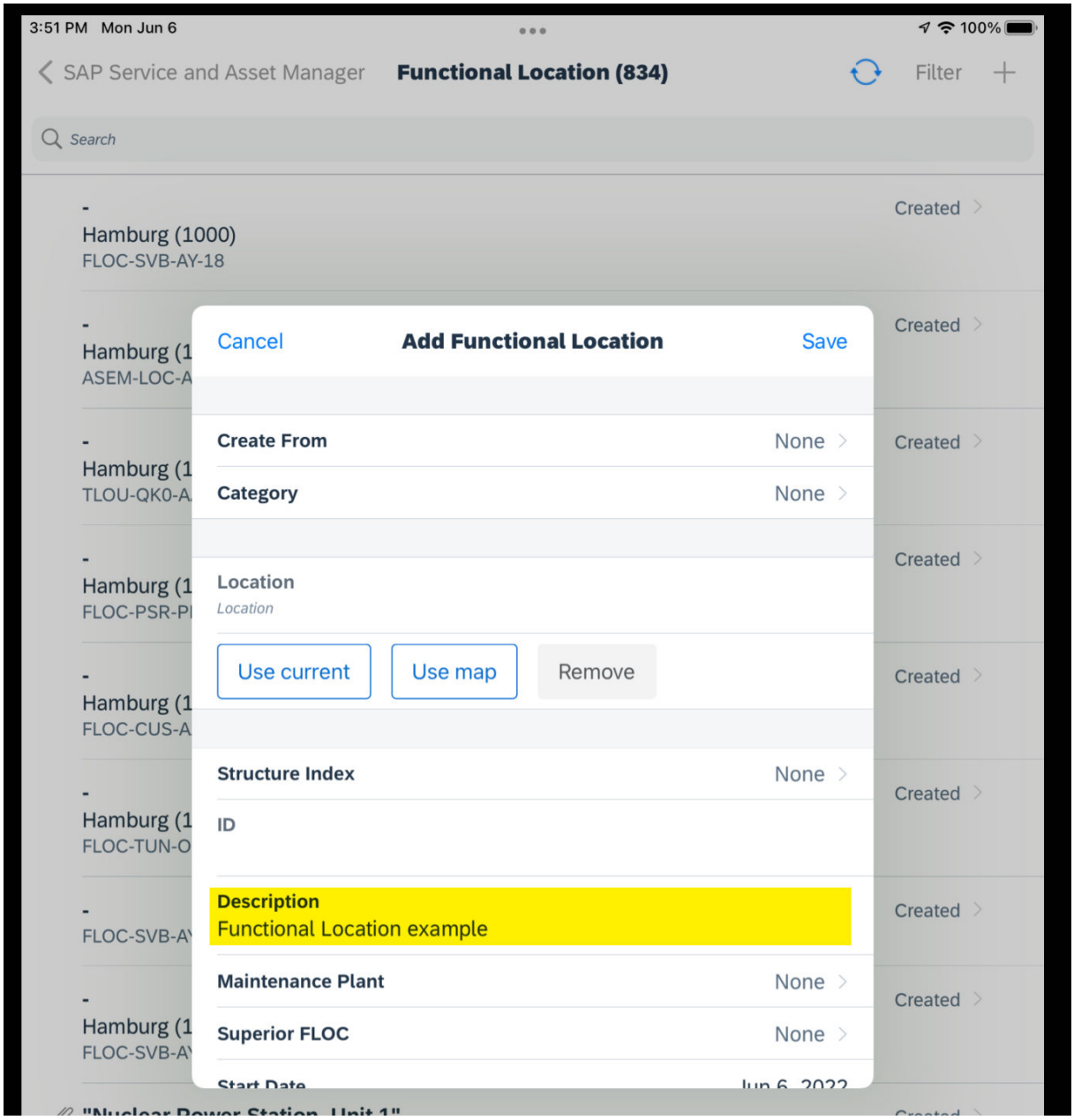


5. Save your changes.

3.2.14.1 App Customization with Additional Fields

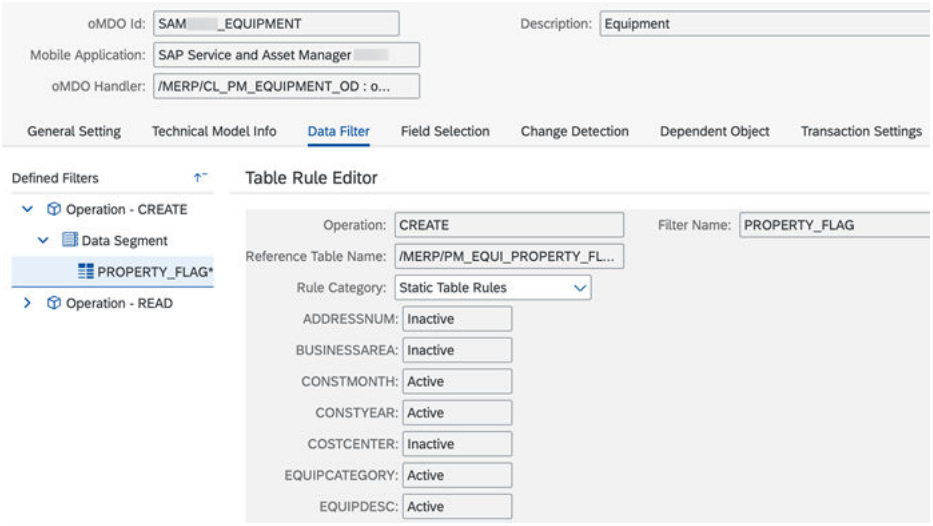
The PROPERTY_FLAG filter in the technical objects, either *Equipment* or *Functional Location* oMDO contains the list of properties available to edit during the creation of a technical object. The properties, which are active by default correspond to the default fields in the applications technical object create page, see the example.



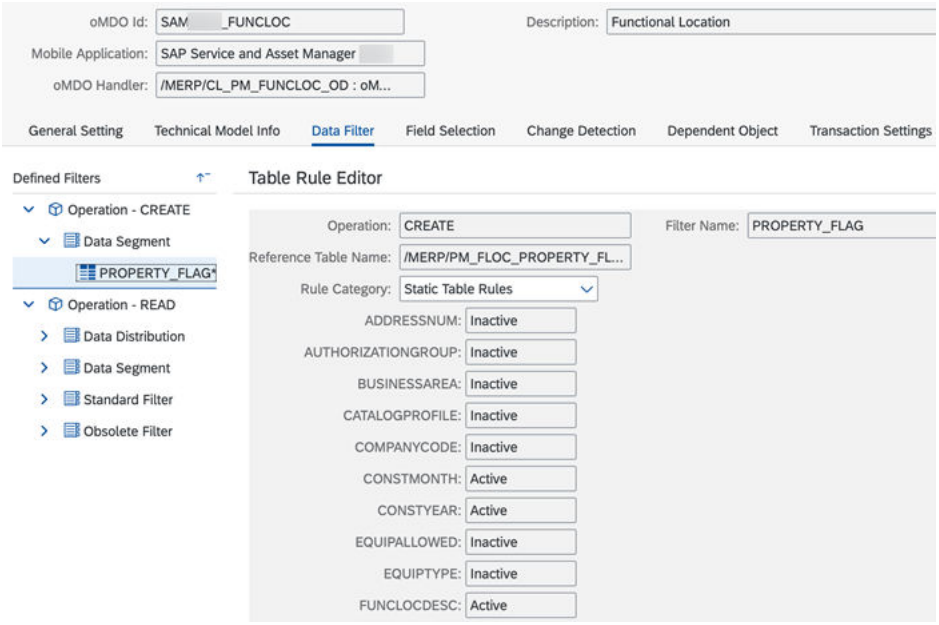


If a property is inactive, the value from the template or a previously created technical object is written to the newly created object. Updating the property to active in the filter doesn't change the create page of the application, a customization would be required in the application, too.

EQUIPDESC corresponds to the *Description* field in the *Equipment* create page of the app.



FUNCLOCDESC corresponds to *Description* field in the *Functional Location* create page of the app.



3.2.15 Auto-Resize Images with a Configured Maximum Size

You can configure the maximum image size, which is allowed for uploading images from the application. Uploaded images exceeding the size limit will be resized automatically.

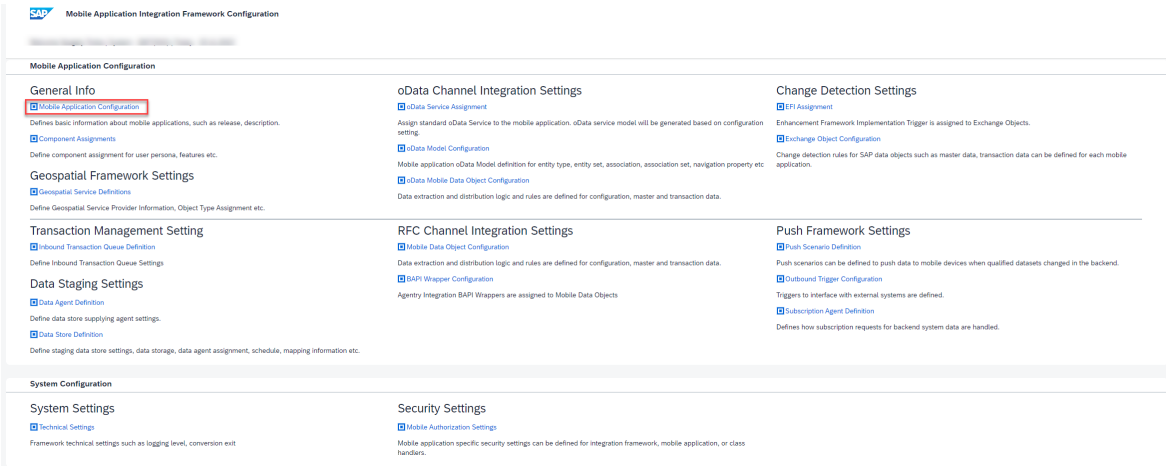
Context

> **Note**
>
> Decreased size of the images will be less, but not necessarily equal to the configured maximum size.

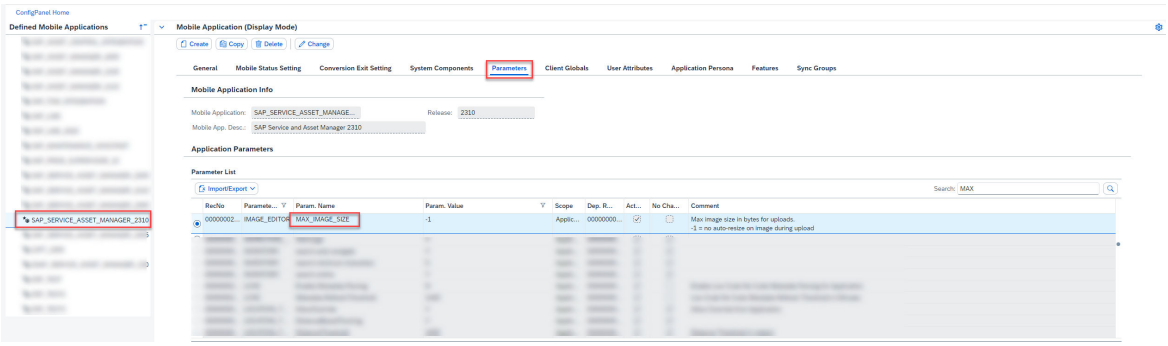
See configuration steps as follows:

Procedure

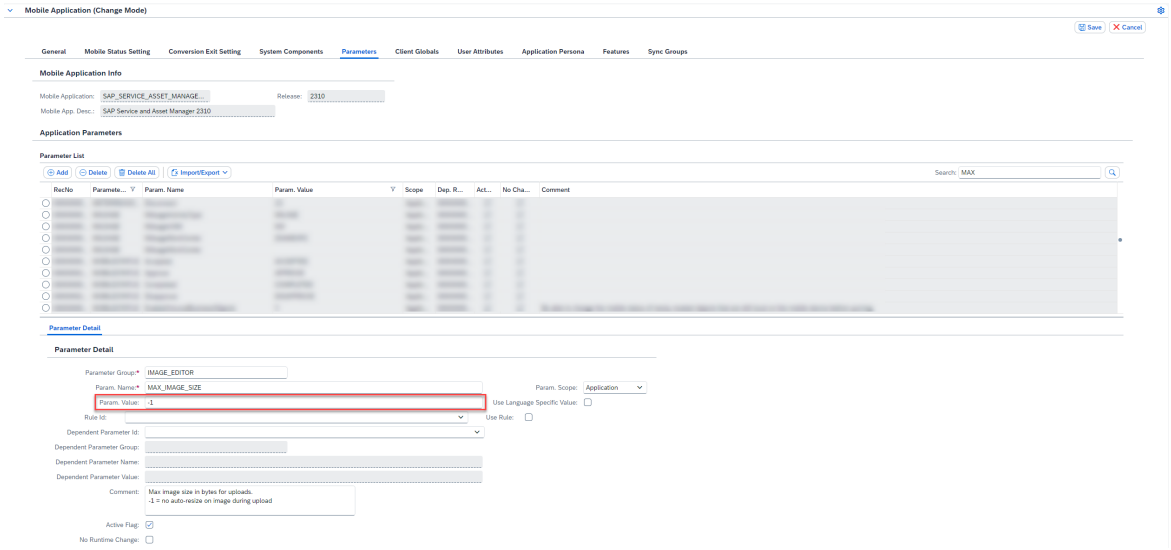
1. On the *Configuration Panel*, go to *Mobile Application Configuration*.



2. Select your mobile application, go to the *Parameters* tab and select the parameter *MAX_IMAGE_SIZE*.



3. Enter *Change mode* and in the *Parameter Details* section add your preferred maximum image size as the *Parameter Value*.



> **Note**
>
> - Add value number in **BYTES**.
> - The default value is **-1**, in this case the auto-resize feature is disabled.

4. Click *Save* to exit *Change mode*.

3.3 OData Channel Integration Settings Procedures

3.3.1 Copying an Object to the Customer Namespace

When you modify either an oData mobile data object or an exchange object, first make a copy of the object and place it in the customer namespace.

Context

The following procedure provides information on making a copy of an oData mobile data object (OMDO) or exchange object within SAP Mobile Add-On. In any of the procedures provided in this guide where an OMDO or an exchange object is copied, refer to this procedure for instructions. When you copy either an OMDO or an exchange object, you can roll back any changes you make to the application if necessary without changing the original objects.

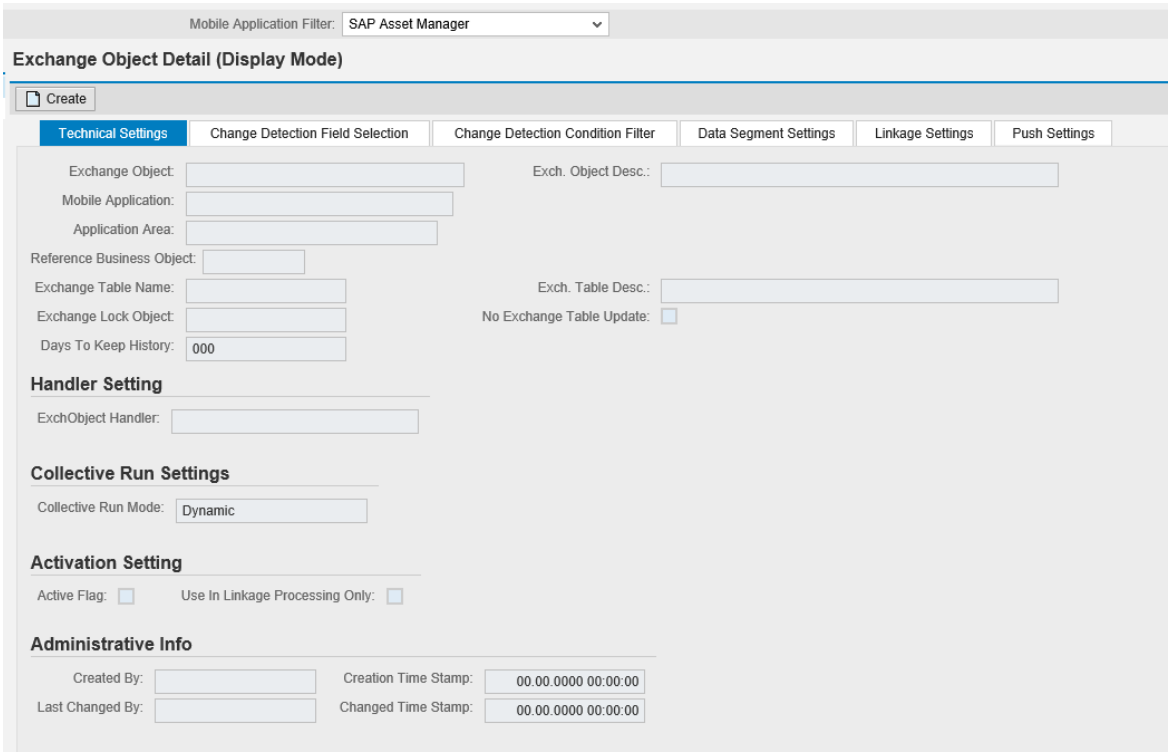
Once you copy an OMDO and modify the object, you may adjust the oData model definition to reference the new OMDO. Similarly, when you copy and modify an exchange object, you may need to change the EFI trigger assignment to the new exchange object. These procedures are covered separately.

Procedure

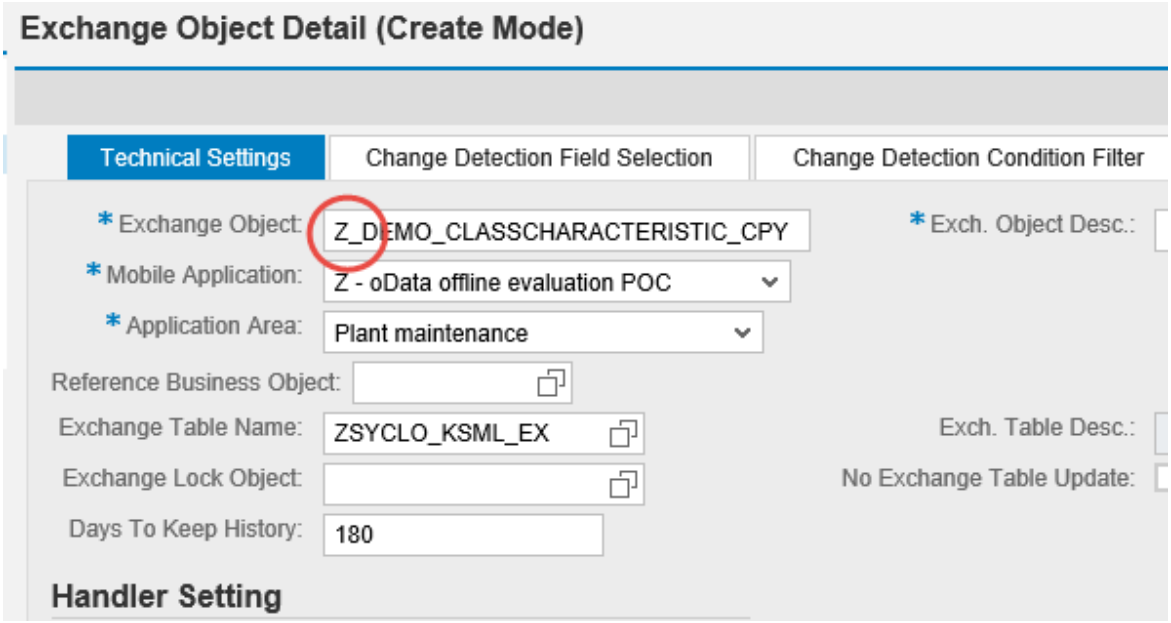
1. Log into the ConfigPanel of the SAP Mobile Add-On.
2. Click either *Exchange Object Configuration* or *oData Mobile Data Object Configuration* from the home page.

   The Object Detail panel opens.

> ### Note
>
> Figures shown in this procedure are taken from the Exchange Object configuration page. Screens may look different when configuring an oData mobile data object. For either, the ability to copy is provided.



3. Select the object to copy from the list of OMDOs or exchange objects and click *Copy*.
4. In the main object `<ID>` field, add a `<Z>` to the beginning of the object name.



5. Click *Save* to save the object copy.

   A copy of the original object is created in the customer namespace. Now you can modify the object, with the original object as a back-up for rollback purposes, if necessary.

3.3.2 Working with oData MDO Filter Rules

Filter rules specify a single field within the database tables from which data is retrieved. Filter rules also specify under which conditions records are included in the operation based on the value of the field.

Data filters are part of the configuration of an oMDO. If you make configuration changes to SAP Service and Asset Manager, you may need to adjust the rules for one or more of the oMDO filters.

Many of the filters in SAP Service and Asset Manager either do not contain active rules or contain rules that you can adjust. A filter only effects the synchronization behavior when it has one or more active rules.

The following procedure instructs you on how to adjust a filter using the ConfigPanel.

3.3.2.1 Changing oData MDO Filter Rules

Many of the common configuration changes made for an SAP Service and Asset Manager implementation involve modifying or adding one or more filter rules in an oData MDO.

Context

In SAP S/4HANA, each user is assigned a role based profile with authorization permissions on viewable data and available activities. For example, a user working in one plant should not be able to view data for a different

plant. When business activities performed by a user are mobilized through the mobile application, the ability to extend the same restrictions to the mobile application is necessary. Data filter rules provide the function to restrict data access for mobile applications.

Use the following procedure to modify a data filter rule for an oMDO. The changes you make to the settings of a given rule vary depending on your mobile application implementation requirements. Subsequent procedures in the *Configuration Guide* refer to this procedure and provide detailed values and settings for filter rules involved in the specific change.

Procedure

1. Access the ConfigPanel. See Accessing the SAP Mobile Add-On for SAP Configuration Panel [page 14] for information.
2. From the ConfigPanel *Home* page, click the *oData Mobile Data Object Configuration* link.
3. At the top of the *oData Mobile Data Object Configuration* page display, in the *Mobile Application Filter* field, choose your mobile application from the dropdown menu. Choosing your mobile application is not a necessary step, but it eliminates objects that are not part of your mobile application from the object list.
4. Click the *Data Filter* tab.
5. Expand the *oData Mobile Data Object List* tree so you can see all of the oData mobile objects.
6. Select the oData mobile data object that requires filter modification from the list.

   The current rule filter settings are displayed in the *Rule Editor* section. All existing rules for the filter are displayed in the *Rule List* table.
7. To add a new rule, edit an existing rule, or delete a rule from the filter, click *Change*.

   Many of the fields in the rule editor become editable, and the buttons *Add Row* and *Delete Row* appear.
8. Set or modify any editable fields desired according to your mobile application needs. For a detailed description of all oData mobile data object fields, see the *OData Channel Integration Settings* topic and and the related subtopics in the section.
9. Set the *Active Flag* to `<True>` for each added or edited field before saving changes. Inactive filter rules have no effect on synchronization processing.
10. Click *Save* to apply your changes.

3.3.3 Work Order Distribution by Order Type Overview

In the default configuration of SAP Service and Asset Manager, work orders are distributed to technicians based on basic parameters. Your site may wish to distribute work orders to users based on the order type.

By default, all Plant Maintenance specific order types are included in the synchronization logic for the SAP Service and Asset Manager application.

In many environments, one or more order types are added to SAP Mobile Add-On specifically for work orders that are distributed to technicians. The added order types indicate that SAP Service and Asset Manager will

only download certain specified work orders. To support this distribution method, change the data filter rules of the OMDOs involved in work order synchronization. The OMDOs include:

- **SAM2310_ORDER_TYPE**
- **SAM2310_WORK_ORDER_GENERIC**

3.3.3.1 Configuring Work Order Distribution by Work Order Type

Creating rules based on work order types affects synchronization processing and work order downloads to the mobile devices of your users.

Prerequisites

Address the following before performing the procedure:

- The order types for work orders that are downloaded to technicians using the SAP Service and Asset Manager application are already determined.
- The person performing the procedure has access to the Config Panel and permissions to change settings.

Context

The following procedure modifies the synchronizing behavior of the SAP Service and Asset Manager application so only work orders with a given order type or types are downloaded to the client. In the procedure, you'll change the ORDER_TYPE filter in the OMDOs involved in work order synchronization. Specifically, you add rules to the filter in each OMDO to include only the desired work order types. You add a rule for each order type to include.

If you don't create a rule for a work order type, then those work order types are excluded from the work order download synchronization processing. If the work orders are excluded from the synchronization processing, then the work orders aren't present on the mobile clients of your users.

Procedure

1. From the Config Panel home page, click the *OData Mobile Data Object Configuration* link, then click the *Data Filter* tab. Be sure to have your desired mobile application chosen in the *Mobile Application Filter* field at the top of the page.
2. Expand the *OData Mobile Data Object by Mobile App* list on the left and click **SAM2310_ORDER_TYPE**.
3. Expand the *Standard Filter* in the *Defined Filters* pane, and click the *ORDER_TYPE* filter.
4. View the rule list for the filter, which is empty in the default configuration of SAP Service and Asset Manager. Click the *Change* button.

5. Create a rule for each order type included in the work order distribution to the SAP Service and Asset Manager technicians. The settings for the rule are as follows:
   - **DOF Rule Type:** Static Value in Range Format
   - **Sign:** Inclusive
   - **Option:** =
   - **Low Value:** The desired order type

   For more details on adding or editing filter rules, see Changing oData MDO Filter Rules [page 149].
6. *Save* your changes once you're finished.
7. Find and click the **SAM2310_WORK_ORDER_GENERIC** OData mobile data object on the list on the left.
8. Expand the *Operation - READ* > *Data Distribution* in the *Defined Filters* pane, and click the *ORDER_TYPE* filter.
9. View the rule list for the filter, which is empty in the default configuration of SAP Service and Asset Manager. Click the *Change* button.
10. Create a rule for each order type included in the work order distribution to the SAP Service and Asset Manager technicians, as you did with the previous OMDO filter. The settings for the rule are as follows:
    - **DOF Rule Type:** Static Value in Range Format
    - **Sign:** Inclusive
    - **Option:** =
    - **Low Value:** The desired order type
11. *Save* the changes.

Results

After you finish the procedure, work orders are downloaded by the SAP Service and Asset Manager application only if their work order type is set to a type for which a filter rule was created. Other work order types aren't retrieved by the application.

3.3.4 Business Object Distribution by Assignment Model

Business object distribution defines the data that needs to be downloaded to the mobile device based on the resource planning of technicians for different business objects, such as work order and notification. You can use this configuration to define which technicians has to complete which activities on the mobile device.

Implementation environments in different business industries or business types may use a different business object model from the default to determine the proper technician assignment for a business object such work order and notification.

3.3.4.1 Work Order Assignment Type Options

By default, the SAP Service and Asset Manager application determines the assignment of a work order using the personnel number of the work order header. However, you can make minor configuration changes to support several work assignment models.

For some customers using Assignment Type 3 for work orders, viewing a list of suboperations is more important than viewing a list of operations. Work order headers are still visible. You can configure your preference using the ConfigPanel for SAP Service and Asset Manager.

For assignment types 2 and 6, some customers prefer the ability to view all operations rather than all work orders. Work order headers are still visible. You can configure your preference using the ConfigPanel for SAP Service and Asset Manager.

Implementation environments in different business industries or business types may use a different business model from the default to determine the proper technician assignment for a work order.

The following assignment types are supported with minor configuration changes:

> **Note**
>
> The SAP HR module is needed for Assignment Types 1–4.

- **Assignment Type 1:** Header-level person responsible for the work order (default, no change required)
- **Assignment Type 2:** Operation-level personnel number of the work order
- **Assignment Type 3:** Sub-operation-level personnel number of the work order
- **Assignment Type 4:** Capacity requirement personnel assignment
- **Assignment Type 5:** Header-level planner group*
  Prerequisite: Mobile user has to have the user parameter *IHG* set up in the user profile parameter.
- **Assignment Type 6:** Operation- or task-level work center*
  Prerequisite: Mobile user has to have the user parameter *AGR* set up in the user profile parameter.
- **Assignment Type 7:** Header-level business partner*
- **Assignment Type 8:** Header-level of the work center*
  Prerequisite: Mobile user has to have the user parameter *VAP* set up in the user profile parameter.
- **Assignment Type A:** Multi resource scheduling (MRS)
- **Assignment Type Z:** Other (custom BAdI option)*

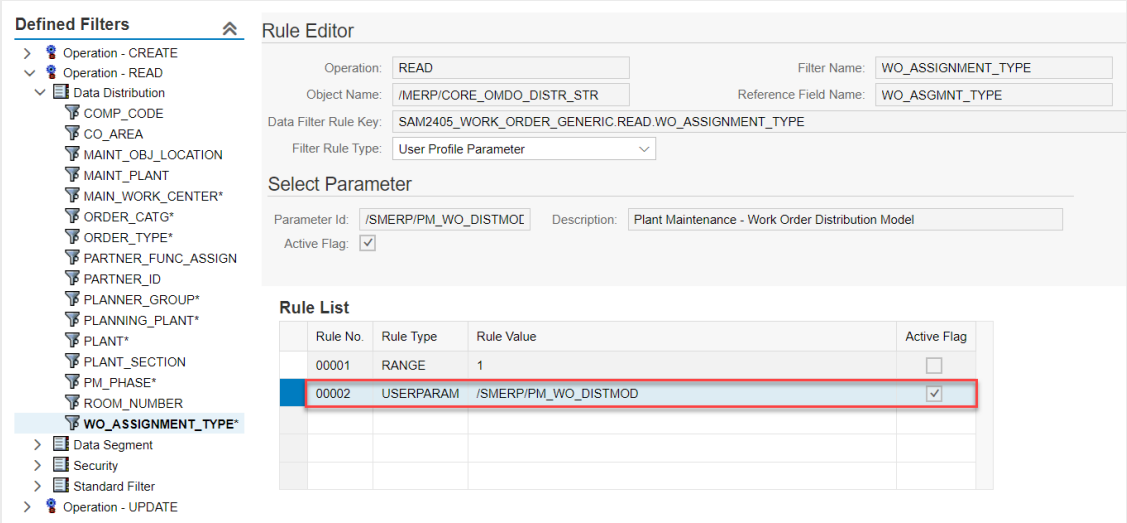
** These assignment types don't require a technician to have a personnel number.*

Perform the following steps to change the assignment type used in a deployment:

1. On the ConfigPanel home page, select *OData Mobile Data Object Configuration*. Make sure to select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. In the *OData Mobile Data Object List* select **SAM2310_WORK_ORDER_GENERIC**, and then the *Data Filter* tab.
3. Expand the *Defined Filters* list as follows; *Operation - READ* > *Data Distribution* > and click *WO_ASSIGNMENT_TYPE*. Click the *Change* button.
4. Set *Low Value* with the desired assignment type as defined by the assignment type model.
5. *Save* your changes.

   > **Note**
   >
   > Please note the following:

- If you are configuring an operation level assignment type, you must update the *OPER_EXCL_SYST_STAT* filter with the *I0009 - CNF:Confirmed* value. However, remove the value if you're configuring a header level assignment type.
- Under *Rule List*, you can set any number of Assignment Types for which the objects will be downloaded from the back end. However, on the client side **only one** of them will be considered by standard, namely the first one on top of the list, which has the Active Flag enabled (see image below). The business logic of this one Assignment Type will be applied to all downloaded objects.
  - Also note that in Supervisor Mode (supported Assignment Types: 6 and 8), you may not see the complete list of objects downloaded from the back end due to filtering. For more information, see Work Order Assignment Type Configuration [page 167].



3.3.4.2 Notification Assignment Options

By default, the SAP Service and Asset Manager application determines the assignment of a notification associated with the notification header. However, you can make minor configuration changes to support several other assignment models for the notification object.

The following assignment types are supported for the notification object:

> **Note**
>
> The SAP HR module is required for *Assignment Type 1* and *Assignment Type 2*.

- **Assignment Type 1:** Header-level person responsible for the notification assignment (default, no change required)
- **Assignment Type 2:** Task-level personnel number of the notification assignment
- **Assignment Type 3:** Header-level planner group*
  Prerequisite: Mobile user has to have the user parameter *IHG* set up in the user profile parameter.
- **Assignment Type 4:** Header-level business partner*
- **Assignment Type 5:** Header-level of the work center*
  Prerequisite: Mobile user has to have the user parameter *AGR* set up in the user profile parameter.

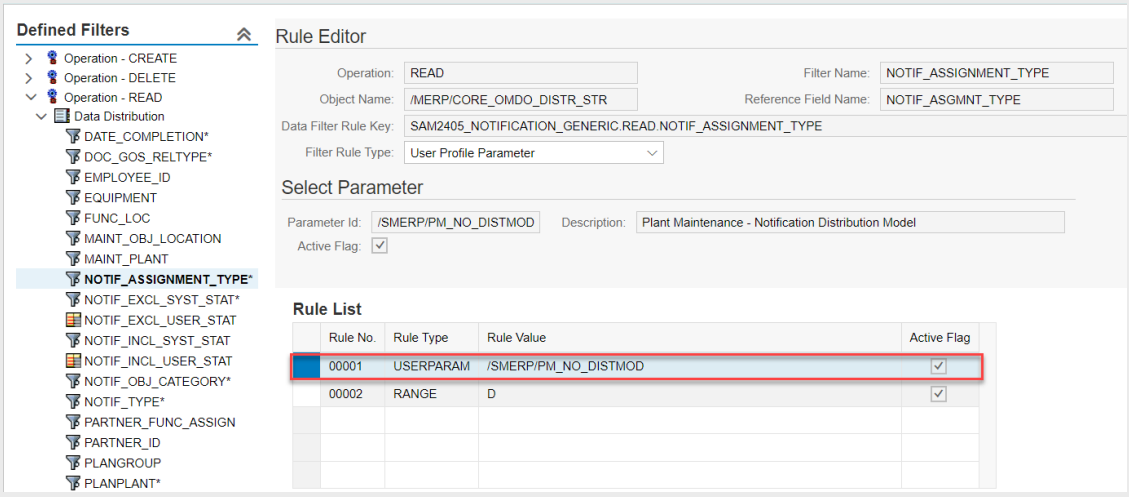
- **Assignment Type D:** Dependent Queue
  By default, this assignment is based on the technician's notification assignment dependent collection*.
- **Assignment Type Z:** Other (custom BAdI option)*

** These assignment types don't require a technician to have a personnel number.*

Perform the following steps to change the assignment type used in a deployment:

1. On the ConfigPanel home page, select *OData Mobile Data Object Configuration*. Make sure to select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. In the *OData Mobile Data Object List* select **SAM2310_NOTIFICATION_GENERIC**, and then the *Data Filter* tab.
3. Expand the *Defined Filters* list as follows; *Operation - READ* > *Data Distribution* > and click *NOTIF_ASSIGNMENT_TYPE*. Click the *Change* button.
4. Set *Low Value* with the desired assignment type as defined by the assignment type model.
5. *Save* your changes.

> **Note**
>
> Please note that under *Rule List*, you can set any number of Assignment Types for which the objects will be downloaded from the back end. However, on the client side **only one** of them will be considered by standard, namely the first one on top of the list, which has the Active Flag enabled (see image below). The business logic of this one Assignment Type will be applied to all downloaded objects.



3.3.5 Filtering Equipment Records by Equipment Status

A large set of records could affect performance on the SAP Asset Manager client. Therefore, you can employ more filtering based on the status of equipment.

By default, SAP Asset Manager filters records through a user-dependent rule based on the planning plant of the user.

To filter records on the status of equipment retrieved for the table stored on the SAP Asset Manager client, modify the **SAM2310_EQUIPMENT** OMDO. Specifically, in the following procedure, you will configure the *EQUI_INCL_SYS_STAT* filter with a rule that specifies which status or statuses to include. After you configure

the rule, only the equipment records with the specified statuses are retrieved by the application for download to the clients.

A common equipment status is *INST*. However, the INST status is only one example of many options. You can configure other filters, either with this example, or in place of it.

For your given SAP Asset Manager implementation, thoroughly review the equipment data stored in the database before deciding which filter rules to configure. After your equipment review, create the appropriate filters within the **SAM2310_EQUIPMENT** OMDO.

3.3.5.1 Configuring an Equipment Status Filter for an Equipment Table

Prerequisites

Address the following items before performing the procedure:

- Know the status or statuses that you are filtering on for equipment synchronization, as they are used in the procedure
- Have access to the ConfigPanel and permissions to change configuration settings

Context

Use the following procedure to create a filter rule for the OMDO, **SAM2310_EQUIPMENT**. Specifically, you are adding a rule to the filter *EQUI_INCL_SYST_STAT*. After you add the filter rule, only the equipment records that match the ones configured in the rule are downloaded to the SAP Asset Manager client.

Procedure

1. Navigate to *ConfigPanel Home* > *OData Mobile Data Object Configuration*.
2. Select your application in the *Mobile Application Filter* field.

   Selecting an application filters the *OData Mobile Data Object by Mobile App* choices in the left panel with only OMDOs available in your application.
3. View the new OMDO copy by selecting it in the *OData Mobile Data Object by Mobile App list*.
4. Select the *Data Filter* tab.
5. In the *Defined Filters* list, click the *Operation - READ* > *Standard Filter* > *EQUI_INCL_USER_STAT* node.
6. Add a rule to the filter with the following configuration settings:
   - **Filter Rule Type:** Static Value in Range Format
   - **Sign:** Inclusive
   - **Option:** =

- **Low Value:** Equipment status to filter on
- **Active Flag:** Checked

7. Repeat the previous step to include additional statuses in the filter.
8. *Save* your changes.

Results

When you finish the procedure, the equipment records downloaded by the SAP Asset Manager application are filtered to only include records with the status or statuses configured in the filter rules.

Next Steps

You may need to filter equipment according to additional criteria. Test that the status filters created during this procedure are performing as expected before creating additional filters for the same data set. Regardless of additional changes, test the synchronization of the equipment data thoroughly after you modify the application.

3.3.6 Retrieving Additional Data for OData Mobile Data Objects

The default implementation of SAP Asset Manager includes the typical data values required by most users and at most implementation. However, it is a common requirement that additional values are retrieved and stored.

3.3.6.1 Adding New Retrievable Values for OData Mobile Data Objects

Prerequisites

Address the following before performing this procedure:

- Determine and note the field values as well as any table values you want to add, as well as which tables the desired fields reside in SAP Mobile Add-On
- You must have access to the ConfigPanel and permissions to change configuration settings within it

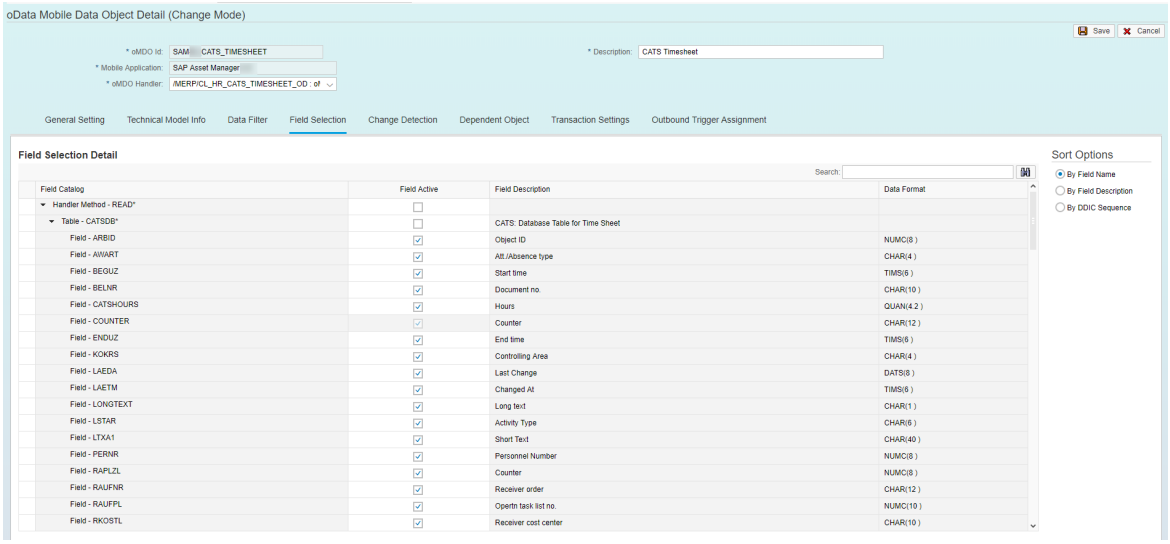
Context

Use the following procedure to add new fields to OData mobile data objects.

Procedure

1. Navigate to *ConfigPanel Home* > *OData Mobile Data Object Configuration*. Select the desired OMDO from the list on the left of the current configuration page.
2. Click the *Field Selection* tab, then click the *Change* button.

   The *Field Active* column is enabled.



3. In the *Field Selection Detail* pane, expand the *Handler Method* and the *Table* to search for the fields that you wish to activate. Fields that are already active are marked with a check in the checkbox.
4. Check the fields that you wish to activate and uncheck any active fields you wish to deactivate. *Save* your changes when finished.

Results

After completing the procedure, one or more new values are retrieved as part of the data for the object. The new values are displayed, edited, searched on, or used in other manners on the mobile client.

In the example screenshot in the procedure, the OData mobile data object used is **SAM2310_CATS_TIMESHEET**. To make other OMDO configuration changes to the object, navigate to the ConfigPanel home page, then click the *OData Model Configuration* link. On the left panel, find the corresponding *EntityType* to make any additional configuration changes. In this procedure example, the entity type is *CatsTimesheet*. See Setting up an OData Mobile Data Object [page 239] for more information.

3.3.7 Enabling or Disabling Follow-On Work Orders

By default, follow-on work orders are enabled in a new installation. You can configure if a follow-on link exists or doesn't exist at the time of a new work order creation.

> **Note**
>
> Ensure that business function *Enterprise Asset Management Part 7 (LOG_EAM_CI_7)* is activated in the back end. See Creating Follow-On Orders for more details.

3.3.7.1 Enabling or Disabling a Follow-On Work Order

Context

Procedure

1. Access the ConfigPanel through SAP Mobile Add-On.
2. Click the *OData Model Configuration* link from the home page of the ConfigPanel.
3. Select and expand the *OData Model Entity Type List* at the left of the pane. Select and expand your application from the list.
4. Select *MyWorkOrderHeader*. Navigate to the *Property List* tab and click the *Change* button.
   a. Delete the *ReferenceOrder* property by clicking the *Delete* button to disable.
   b. Enable the *ReferenceOrder* property by clicking the *Add Property* button and adding the following:
      - **Property Name:** ReferenceOrder
      - **OMDO Field Name:** REFNR (select from a drop-down list)
      - **Edm Type:** Edm.String (select from a drop-down list)
      - **Max Length:** 12
      - **Conversion Exit:** ALPHA
5. *Save* your changes.
6. Click the *OData Mobile Data Object Configuration* link from the home page of the ConfigPanel.
7. Choose your mobile application from the dropdown menu at the top of the *OData Mobile Data Object Configuration* page display, in the *Mobile Application Filter* field. Choosing your mobile application isn't a necessary step, but it eliminates objects that aren't part of your mobile application from the object list.
8. Click the *Field Selection* tab.
9. Expand the *OData Mobile Data Object List* tree so you can see all of the OData mobile objects.
10. Select *SAM2310_WORK_ORDER_GENERIC* from the list. Click the *Change* button.

11. Expand the *Table - AUFK* node, and then locate the *REFNR* field.
12. Enable or disable the field by selecting or deselecting the *Field Active* checkbox.
13. *Save* your changes.

3.3.8 Maintaining Customer Service Order Types

Prerequisites

Be sure that you have installed the Customer Service component. See the instructions in the Asset Manager Component Installation Guide for IOS for more information.

> **Note**
>
> Configuring Customer Service order types is optional and is required only if the Customer Service component is enabled.

Procedure

1. Access the ConfigPanel through SAP Mobile Add-On.
2. From the ConfigPanel *Home* page, click the *OData Mobile Data Object Configuration* link.
3. At the top of the *OData Mobile Data Object Configuration* page display, in the *Mobile Application Filter* field, choose your mobile application from the dropdown menu. Choosing your mobile application is not a necessary step, but it eliminates objects that are not part of your mobile application from the object list.
4. Click the *Data Filter* tab.
5. Expand the *OData Mobile Data Object List* tree so you can see all of the OData mobile objects.
6. Select *SAM2310_WORK_ORDER_GENERIC* from the list. Then select *Data Filter tab* > *Operation - READ* > *Data Distribution* > *ORDER_TYPE*

   The current rule filter settings are displayed in the *Rule Editor* section. All existing rules for the filter are displayed in the *Rule List* table.
7. To activate the Customer Service order type, click *Change*.

   Many of the fields in the rule editor become editable.
8. Set the *Active Flag* to *True* for the `<Rule Value>` */MERP/CL_PM_ORDER_TYPE_ORU?CS*.
9. To apply your changes, click *Save*.

Results

The Customer Service order type is activated in the ConfigPanel.

3.3.9 Maintaining Customer Service Notification Types

Prerequisites

Be sure that you have installed the Customer Service component. See the instructions in the Asset Manager Component Installation Guide for IOS for more information.

> **Note**
>
> Configuring Customer Service notification types is optional and is required only if the Customer Service component is enabled.

Procedure

1. Access the ConfigPanel through SAP Mobile Add-On.
2. From the ConfigPanel *Home* page, click the *OData Mobile Data Object Configuration* link.
3. At the top of the *OData Mobile Data Object Configuration* page display, in the *Mobile Application Filter* field, choose your mobile application from the dropdown menu. Choosing your mobile application is not a necessary step, but it eliminates objects that are not part of your mobile application from the object list.
4. Click the *Data Filter* tab.
5. Expand the *OData Mobile Data Object List* tree so you can see all of the OData mobile objects.
6. Select *SAM2310_NOTIFICATION_GENERIC* from the list. Then select *Data Filter tab* > *Operation - READ* > *Data Distribution* > *NOTIF_TYPE*

   The current rule filter settings are displayed in the *Rule Editor* section. All existing rules for the filter are displayed in the *Rule List* table.
7. To activate the Customer Service order type, click *Change*.

   Many of the fields in the rule editor become editable.
8. Set the *Active Flag* to *True* `<Rule Value>` */MERP/CL_PM_NOTIF_TYPE_ORU?CS*.
9. To apply your changes, click *Save*.

Results

The Customer Service notification type is activated in the ConfigPanel.

3.3.10 Configuring Confirmation Posting Dates - Overview

You can use either the *USE_USER_TIME_ZONE* or the *POSTING_DATE* filter, found in the **SAM2310_PM_CONFIRMATION** OMDO, to configure confirmation posting dates. Note that you can enable only one filter at a time. If both filters are enabled, you'll get an error.

USE_USER_TIME_ZONE Filter

See the Configuring Confirmation Posting Date Using the USE_USER_TIME_ZONE Filter [page 163] procedure for detailed instructions.

When the SAP system and mobile users are in different time zones, use the *USE_USER_TIME_ZONE* filter to perform the time zone conversion. When the filter is active, the application uses the time zone of the mobile user in the back end system to convert the actual start date / time. It also converts the finish date / time received from the mobile client. The system time zone is the default setting. The User Time Zone filter is a back end only configuration.

> **Note**
>
> To use the time zone handling functionality in confirmation, SAP system must be customised with time zone. If SAP system time zone isn't maintained and the *USE_USER_TIME_ZONE* filter is active in the **SAM2310_PM_CONFIRMATION** OMDO, posting of confirmation from SAP Service and Asset Manager raises an error.

POSTING_DATE Filter

See the Configuring Confirmation Posting Date Using the POSTING_DATE Filter [page 164] procedure for detailed instructions.

The *POSTING_DATE* filter overrides the confirmation posting date as system date. The default option uses the posting date from the mobile device. If you use the *POSTING_DATE* filter, you must also enable the *PostingDateFromUserOverride* parameter. The Posting Date filter is a combination of front end and back end configuration.

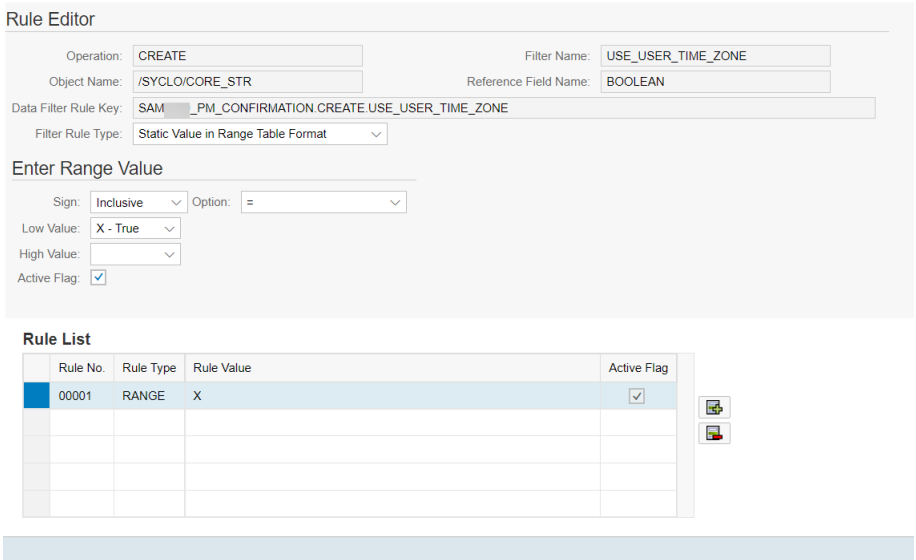
3.3.10.1 Configuring Confirmation Posting Date Using the USE_USER_TIME_ZONE Filter

Context

Use the following procedure if you're configuring the confirmation posting date using the *USE_USER_TIME_ZONE* filter.

Procedure

1. On the ConfigPanel Home page, select *OData Mobile Data Object Configuration*. Make sure to select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. Click on *oData Channel Integration Settings* > *oData Mobile Data Object Configuration*.
3. In the *OData Mobile Data Object List* select **SAM2310_PM_CONFIRMATION**, and then the *Data Filter* tab.
4. Expand the *Defined Filters* list as follows: *Operation - CREATE* > *Standard Filter* > *USE_USER_TIME_ZONE*. Click the *Change* button.
5. Select the existing rule. Ensure the `<Low Value>` field is set to *X - True* and is enabled:



6. *Save* your changes.

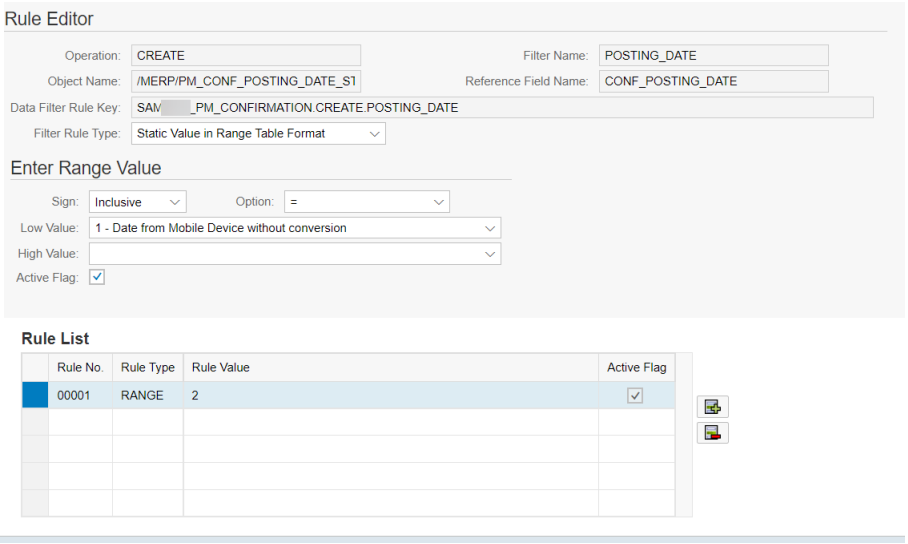
3.3.10.2 Configuring Confirmation Posting Date Using the POSTING_DATE Filter

Context

Use the following procedure if you're configuring the confirmation posting date using the *POSTING_DATE* filter.

Procedure

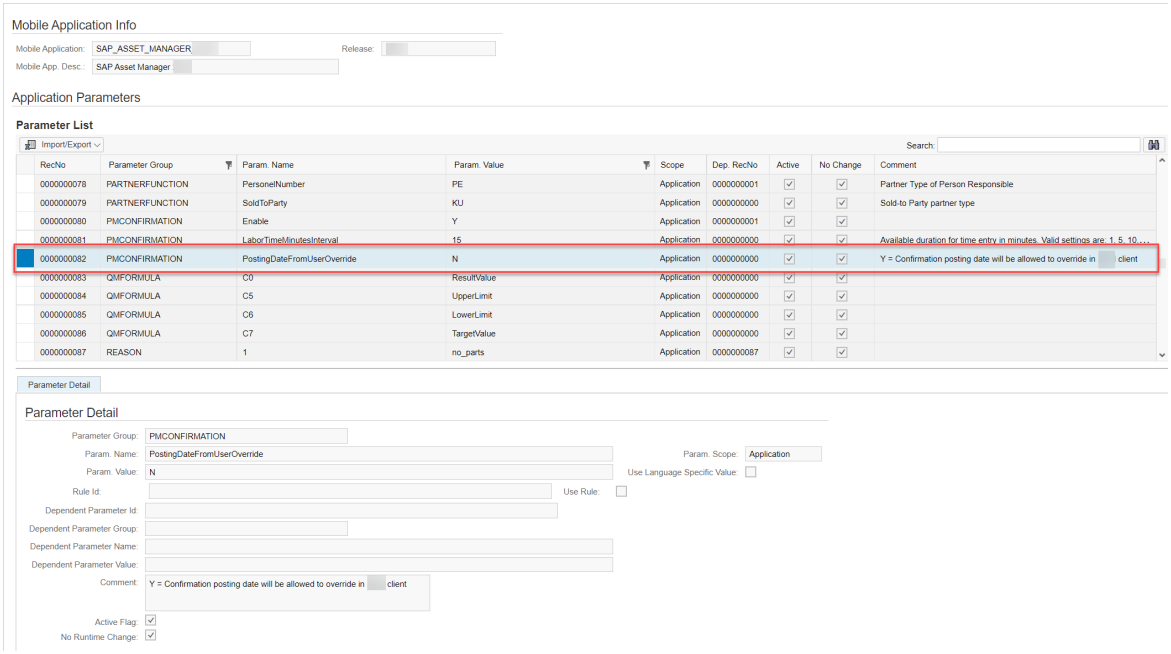
1. On the ConfigPanel Home page, select *OData Mobile Data Object Configuration*. Make sure to select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. Click on *oData Channel Integration Settings* > *oData Mobile Data Object Configuration*.
3. In the *OData Mobile Data Object List* select **SAM2310_PM_CONFIRMATION**, and then the *Data Filter* tab.
4. Expand the *Defined Filters* list as follows: *Operation - CREATE* > *Standard Filter* > *POSTING_DATE*. Click the *Change* button.
5. Select the existing rule. Ensure the `<Low Value>` field is set to *1* and is enabled:
   - **Default:** 1 - Date from Mobile Device without conversion
   - 2 - User Time Zone Date (at the time of BAPI execution)
   - 3 - System Time Zone Date (at the time of BAPI execution)



6. *Save* your changes.
7. Return to the *Home* page. Navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.

8. Click the *Change* button.
9. Find and highlight the *PostingDateFromUserOverride* parameter, located in the *PMCONFIRMATION* parameter group. Set the parameter to one of the following:
    - **N:** (Default). If set to N, the confirmation posting date is automatically taken from the time zone set on the mobile client.
    - **Y:** If set to Y, the mobile client user can manually enter a date on the Confirmation screen in the app.



10. Check the `<Active>` flag to ensure that the parameter is used by the mobile application. If desired, and if not already checked, check the `<No Runtime Change>` box to ensure that the value of the parameter isn't overridden at runtime through synchronization processing.
11. *Save* your changes.

3.3.11 Configuring the Supervisor Module

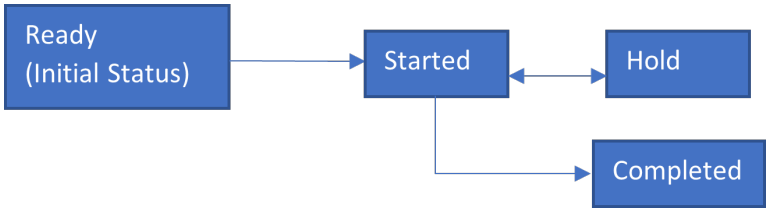
Use the following procedures to configure the supervisor module:

- Work Order Assignment Type Configuration [page 167]
- Supervisor Roles and Team Assignment Type Configuration [page 168]
- Enabling the Supervisor Approval Process for Work Orders [page 170]
- Mobile Application Parameter Settings for the Supervisor Mode [page 171]
- Mobile Application Status Settings for the Supervisor Mode [page 174]
- Rejection Reason Configuration [page 174]

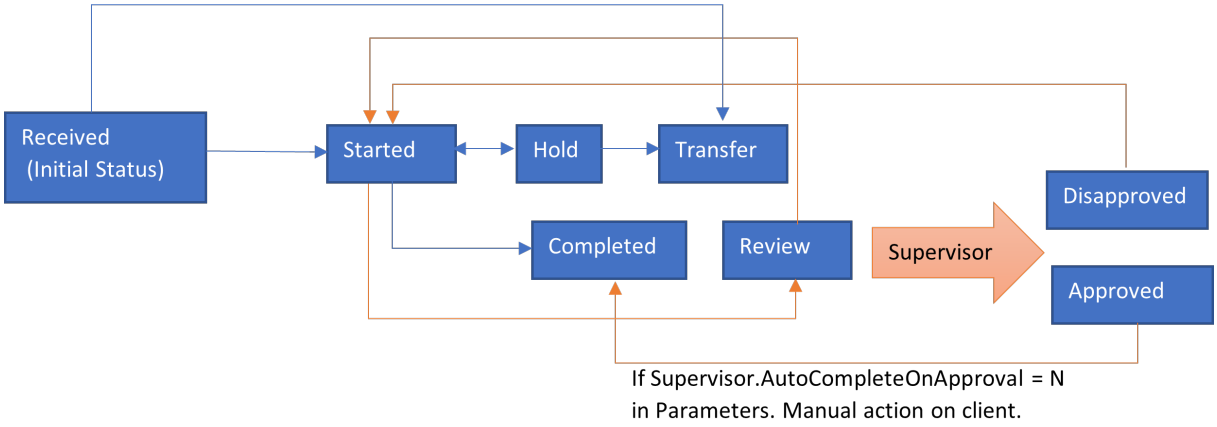
3.3.11.1 Work Order Operations State Machine for Maintenance Technician Persona

Technician Role:

- Phase model relevant operations (Supervisor mode is Active or Inactive)



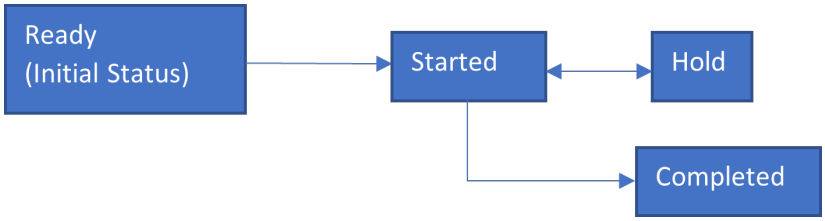
- Non-phase model relevant operations (Supervisor mode is Active)



> **Note**
>
> Allowed status transitions are shown by red lines. Status transitions marked by red are only available when the PM_SUPERVISOR_MODE feature is active. All other transitions are available by default. The supervisor approves or disapproves the operation after it enters the Review mode by a technician.

Supervisor Role:

- Phase model relevant operations (Supervisor mode is Active or Inactive)

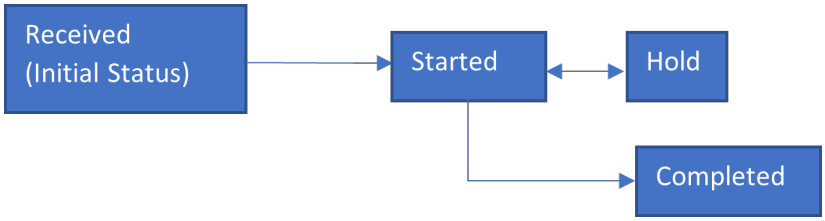


> **Note**
>
> Here the user is a supervisor and is logged in as a supervisor. These state transitions are available even if the PM_SUPERVISOR_MODE feature is inactive. However, a scenario in which the supervisor configuration and authorization in the backend exists without the PM_SUPERVISOR_MODE feature active is unlikely.

- Non-phase model operations

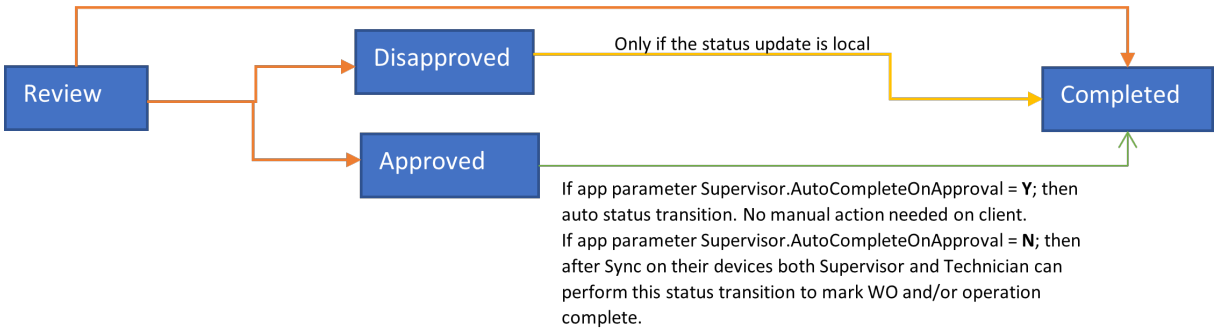
  > **Note**
  >
  > Feature PM_SUPERVISOR_MODE is active. Allowed status transitions are shown by red lines.

Working as a Technician:



> **Note**
>
> Here the supervisor is a technician with additional responsibilities.

Working as a Supervisor:



3.3.11.2 Work Order Assignment Type Configuration

Context

The following assignment types are supported when using the Supervisor module:

- **Supported work order assignment types for the technician**
  - 1 - Header level person responsible

- 2 - Operation/Task level personnel number
- 6 - Operation/Task level work center
- 8 - Header level work center

- **Supported work order assignment types for the supervisor**
  - 6 - Operation/Task level work center
  - 8 - Header level work center

Configure the assignment types for the work order OData MDO as follows:

> **Note**
>
> One of the assignment types must be configured for the technician, while the other assignment type must be configured for the supervisor based on the supported work assignment types. Only the same level of assignment types is supported. In this, you can configure either header level assignments or operation level assignments, and not both.

Procedure

1. On the ConfigPanel home page, select *OData Mobile Data Object Configuration*. Make sure to select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. In the *OData Mobile Data Object List* select **SAM2310_WORK_ORDER_GENERIC**, and then the *Data Filter* tab.
3. Expand the *Defined Filters* list as follows; *Operation - READ* > *Data Distribution* and click *WO_ASSIGNMENT_TYPE*. Click the *Change* button.
4. Set *Low Value* with the desired assignment type as defined by the assignment type model.
5. *Save* your changes.

   > **Note**
   >
   > If you're configuring an operation level assignment type, you must update the *OPER_EXCL_SYST_STAT* filter with the *I0009 - CNF: Confirmed* value. However, remove the value if you're configuring a header level assignment type.

3.3.11.3 Supervisor Roles and Team Assignment Type Configuration

Configure the data filters to define the Supervisor role and Team Assignment types for the OMDO. Your specific configuration is based on how the supervisor team is maintained in the back end.

For detailed instructions on how to work with data filters, see the Working with oData MDO Filter Rules [page 149] and Changing oData MDO Filter Rules [page 149] topics.

You're configuring the data filters found in the **SAM2310_USER_ROLE** OMDO.

In addition to configuring the data filters found in the **SAM2310_USER_ROLE** OMDO, you must manually maintain the team assignments for each user in the Mobile Administration and Monitoring Portal. Information on configuring user parameters is found at the following locations:

- **Mobile Add-On for SAP S/4HANA Security Guide**
  - Defining an Additional User Role Required for a Mobile Application User
  - Defining an Additional User Role Required for an OData Mobile Object Handler
- **SAP Mobile Add-On Security Guide**
  - Defining an Additional User Role Required for a Mobile Application User
  - Defining an Additional User Role Required for an OData Mobile Object Handler

WO_ASSIGNMENT_TYPE Data Filter

The following team assignment types are supported for downloading the supervisor team:

- **1 - Work Center Assignment:** Use when the team (technicians and supervisors) is maintained in the back end through the Work Center Maintenance transaction (CR02/CR03)
- **2 - Organizational Structure Assignment:** Use when the team is maintained in the back end through the Org. Structure Assignment (PPOMA transaction).
- **3 - User Attribute Assignment:** Use when the team isn't maintained in the back end through either the Work Center Maintenance or the Org. Structure Assignment transactions. Additional configuration and manual team assignments are needed.
- **4 - Custom Assignment:** Use by the customer to define custom logic for downloading the teams. Customer logic is implemented in the BADI method `/MERP/CA_OMDO_USER_ROLE_BADI - GET_ASSIGNMENT_OTHERS`.

POSITION_ROLE_TYPE Data Filter

The position role type filter defines the Position Org ID for the supervisor, and is maintained as part of the team configuration in the back end. The definition is needed to identify the mobile user with the Supervisor role. It's also needed for Team Assignment types 1 and 2.

WORKCENTER_ROLE_TYPE Data Filter

The work center role type filter is used for Team Assignment type 3. When using Team Assignment type 3, create a new filter rule with the following properties:

- Rule Type: Non-Static Table Rules
- Non-Static Table Rules: Mobile User Attribute
- User Attribute: WORKCENTER_ROLETYPE
- Active: Checked

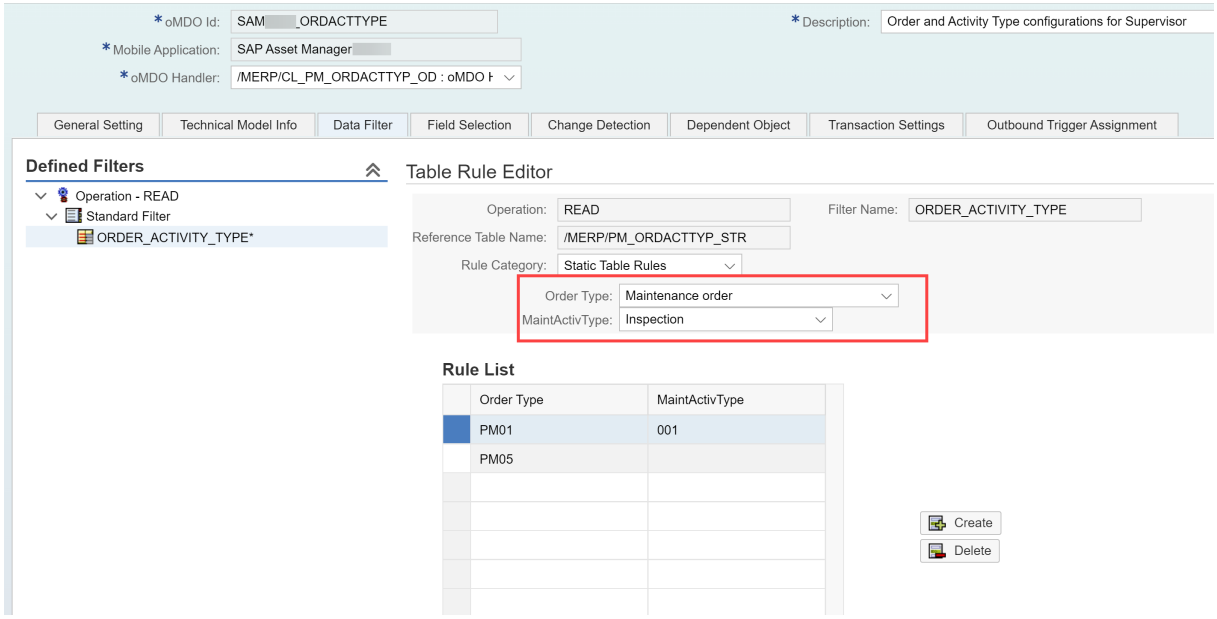
3.3.11.4 Enabling the Supervisor Approval Process for Work Orders

Configure the data filters to define the Supervisor role and Team Assignment types for the OMDO. Your specific configuration is based on how the supervisor team is maintained in the back end.

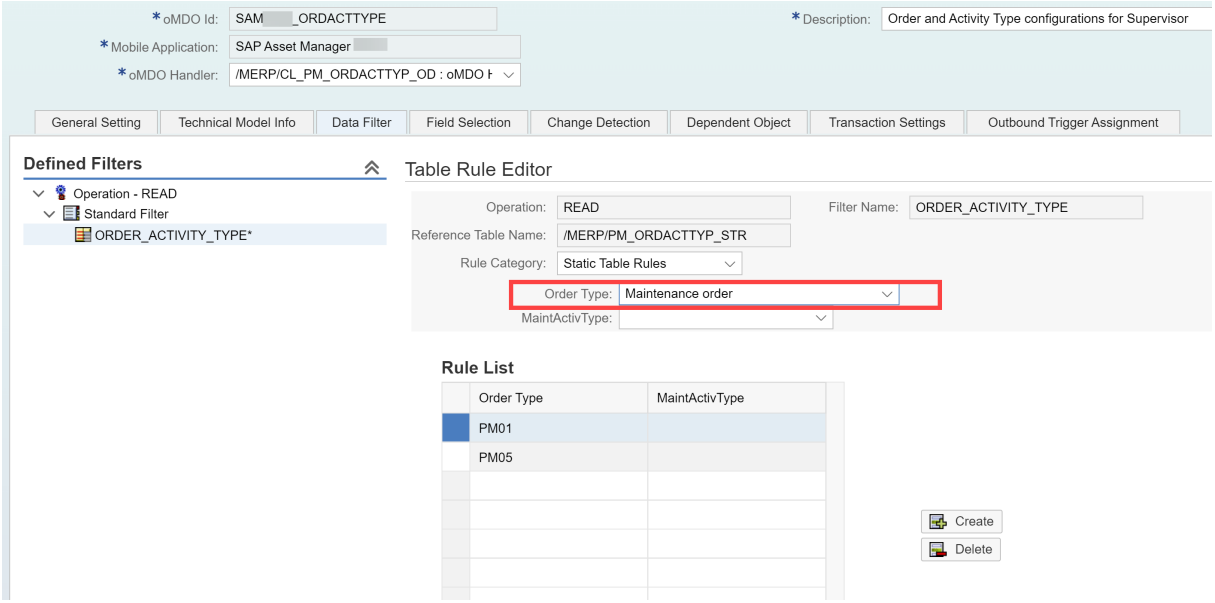
For detailed instructions on how to work with data filters, see the Working with oData MDO Filter Rules [page 149] and Changing oData MDO Filter Rules [page 149] topics.

You're configuring the *ORDER_ACTIVITY_TYPE* data filter found in the **SAM2310_ORDACTTYPE** OMDO.

If `<Order Type>` and `<MaintActiv Type>` fields are specified, the supervisor approval functionality is enabled for the combination:



If only <Order Type> is specified, the supervisor approval functionality is enabled for all orders of that type:



3.3.11.5 Mobile Application Parameter Settings for the Supervisor Mode

Context

> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

> **Note**
>
> As of the SAP Service and Asset Manager 2305 release, SAP introduces a new parameter for the supervisor that allows you to enable auto completion on approval. See the configuration procedure below.

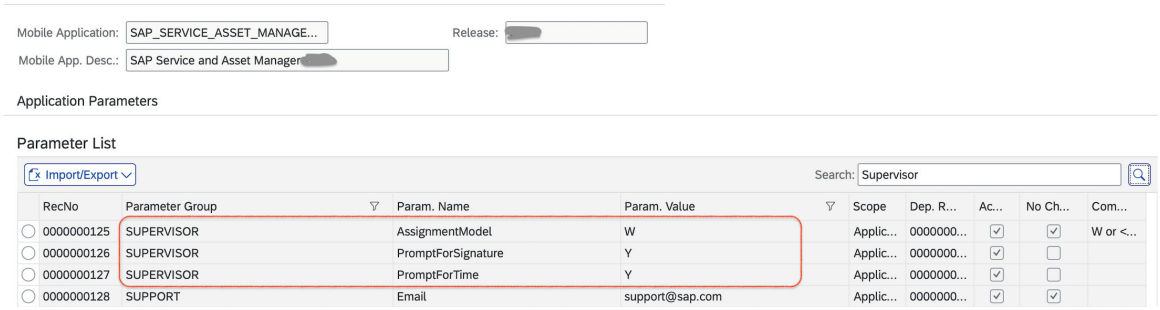
The Supervisor feature is called *PM_SUPERVISOR_MODE*. By default, the feature is not enabled.

To configure the supervisor options for SAP Service and Asset Manager, use the *SUPERVISOR* parameter group and the following parameters within the group:

Procedure

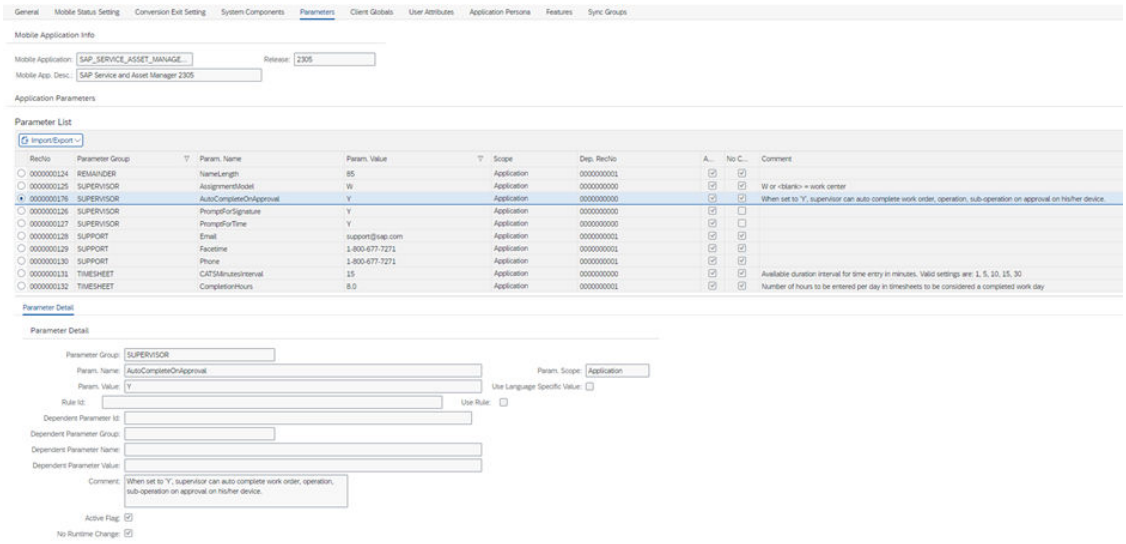
1. Using the ConfigPanel, navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters for the application.
2. The supervisor parameters are found in the *SUPERVISOR* group. You can scroll down to find the parameter, or perform a search using the *Search* box. Highlight the parameter you want to configure and click the *Change* button.



3. Make desired parameter changes as follows:
   a. **AssignmentModel:** Set according to how your team is maintained and based on your team assignment type configuration:
      - For team assignment types *1* or *3*, set the value to *W* or *<blank>*.
      - For team assignment type *2*, set the value to *O*.
   b. **PromptForSignature:** Set the parameter to *Y* if signature capability is needed for supervisor approval.
   c. **PromptForTime:** Set the parameter to *Y* if time capture is needed when the supervisor is executing the approval process.
   d. **AutoCompleteOnArppoval**:

   - If set to *Y*, the supervisor can automatically complete a work order, operation, or sub-operation after approval on his device.
   - If set to *N*, the supervisor can only set the work order, operation, or sub-operation to the *Approved* status.
     After synchronization on their devices, both the assigned technician and supervisor can set the status to *Completed* by manually changing the status on the device.



4. Locate the parameter group *USER_AUTHORIZATIONS*. The parameter *SupervisorRole* is optional and is blank by default. You can configure it in one of the following ways:

> **Note**
>
> If the *Parameter Value* field and the *Rule ID* fields are blank, and if no data filter rule is created, the authorization model is not used. The *UserRoles* entity set is then used to determine the authorization role.

   - Set the *Parameter Value* field:
     - <blank>: The parameter is ignored
     - Y: Supervisor. If set, the current user has the role of Supervisor and the UserRoles entity set is ignored.
     - N: Technician. If set, the current user has the role of Technician and the UserRoles entity set is ignored.
   - Use the authorization check rule */SMFND/CL_CORE_AUTH_CHECK_RU* and corresponding authorization object to check if the user is a supervisor. If the parameter is enabled, it determines if the logged-in user is a supervisor or not in the application. Be sure to check the *Use Rule* box if you're using an authorization check rule.
   - Use the position org ID defined in the *POSITION_ROLE_TYPE* data filter rule in the **SAM2310_USER_ROLE** OMDO. By default, the data filter rule is blank. See Supervisor Roles and Team Assignment Type Configuration [page 168] for information on creating a position role type data filter rule.
5. *Save* your changes.
6. Use the SAP GUI transaction *SU3* to assign parameters depending on assignment types:
   a. **VAP:** Main work center for maintenance tasks. Assign the work center of the user to this parameter when using team assignment type *1* or *3*.
   b. **/MERP/SUPVR_ORGID:** Configure this parameter when using team assignment *2* as follows:
      - For a supervisor, configure with the org ID of the supervisor.
      - For a technician, configure with the org ID of the supervisor, which typically is the same as the org ID of the technician.

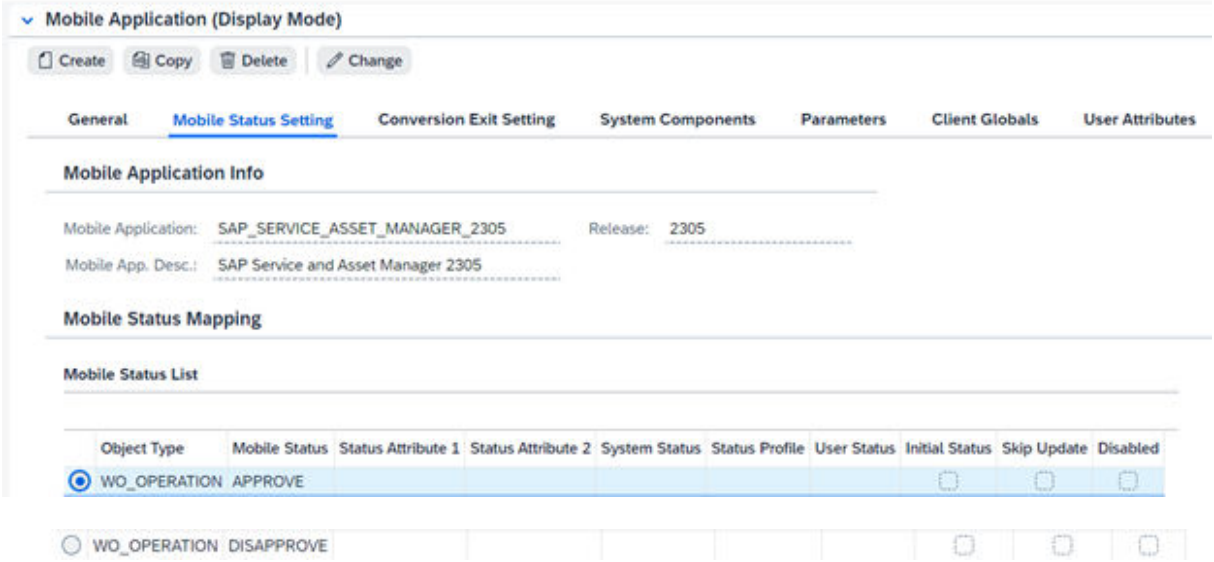
3.3.11.6 Mobile Application Status Settings for the Supervisor Mode

> **Note**
>
> By default status settings for the supervisor mode aren't mapped to any back-end user or system status. For an example of how to configure a mapping status, see the Changing the Mapping of a Mobile Status to STARTED [page 99] procedure.

The following statuses to support the supervisor functionality are available at the work order and operation levels, depending on your configuration of the work order assignment type:

- **REVIEW:** Set by the technician after they've completed their work. The supervisor is then required to review the work.
- **APPROVE** and **DISAPPROVE**

  > **Note**
  >
  > **ACCEPTED** and **REJECTED:** Set by the field service technician if he is rejecting the work done on the work order or operation.



When a supervisor approves a work order or an operation, the status moves to COMPLETED.

3.3.11.7 Rejection Reason Configuration

The supervisor can reject an order or operation as part of the review process. If an order or operation is rejected, the supervisor must specify a rejection reason.

Configure the rejection reasons in the **SAM2205_REJECTION_REASON** OMDO, using the *REJECTION_REASON* data filter.

For information on how to work with data filters, see the Changing oData MDO Filter Rules [page 149] topic.

3.3.12 Configuring EAM Phase Model - Overview

The EAM Phase Model feature is called *EAM_PHASE_MODEL*.

> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

> **Note**
>
> This configuration is designed to work with SSAM with both enabled and unenabled order types. Phase model specific notifications are not supported in SSAM.

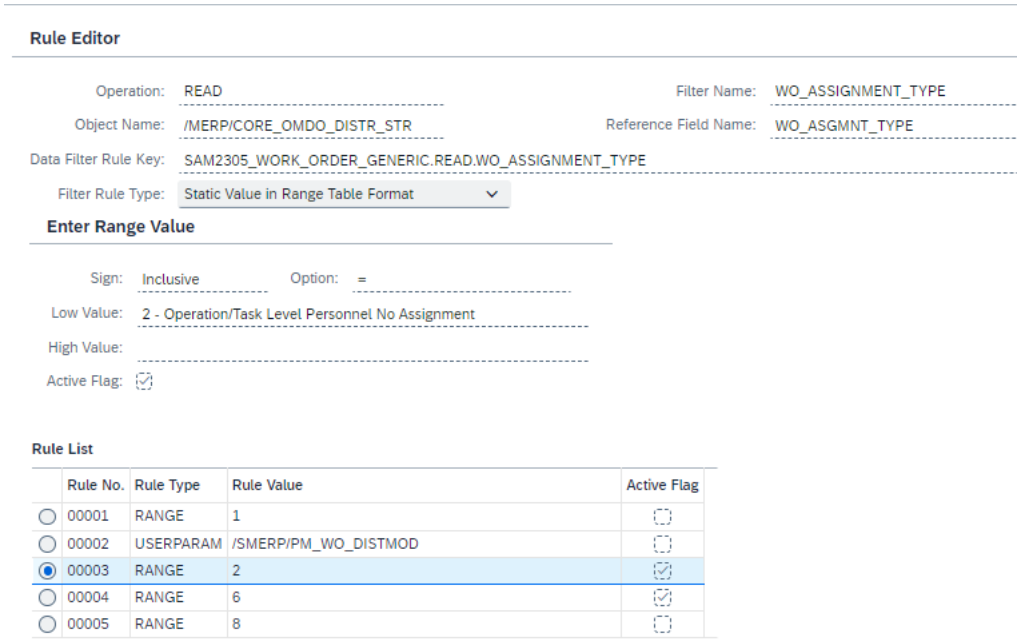
1. Work Order oMDO Configuration [page 175]
2. Notification Assignment Type Configuration [page 180]
3. Order Type oMDO Configuration [page 182] The process control code can limit a certain process state change based on the system state. It can also be applied to the execution phase, which prevents the user from moving the order or operation from one sub-phase to another.
4. Notification Type oMDO Configuration [page 183]
5. Work Order Partner Determination Procedure oMDO Configuration [page 185]
6. Notification Partner Determination oMDO Configuration [page 186]
7. Overall Status oMDO Configuration [page 187]
8. Mobile Status Mapping Configuration [page 189]
9. Mobile Status State Machine [page 190] The ConfigPanel shows you which PM mobile statuses are mapped to phase model statuses.
10. Persona and Feature Enablement [page 192]
11. Minor Emergency Work [page 195]
12. EAM Phase Control Code [page 196]
    This new feature is added to support emergency or minor work in a reactive maintenance phase model. It helps to skip phases and issue a work order or notification when a technician is authorized to create those items.

3.3.12.1 Work Order oMDO Configuration

Context

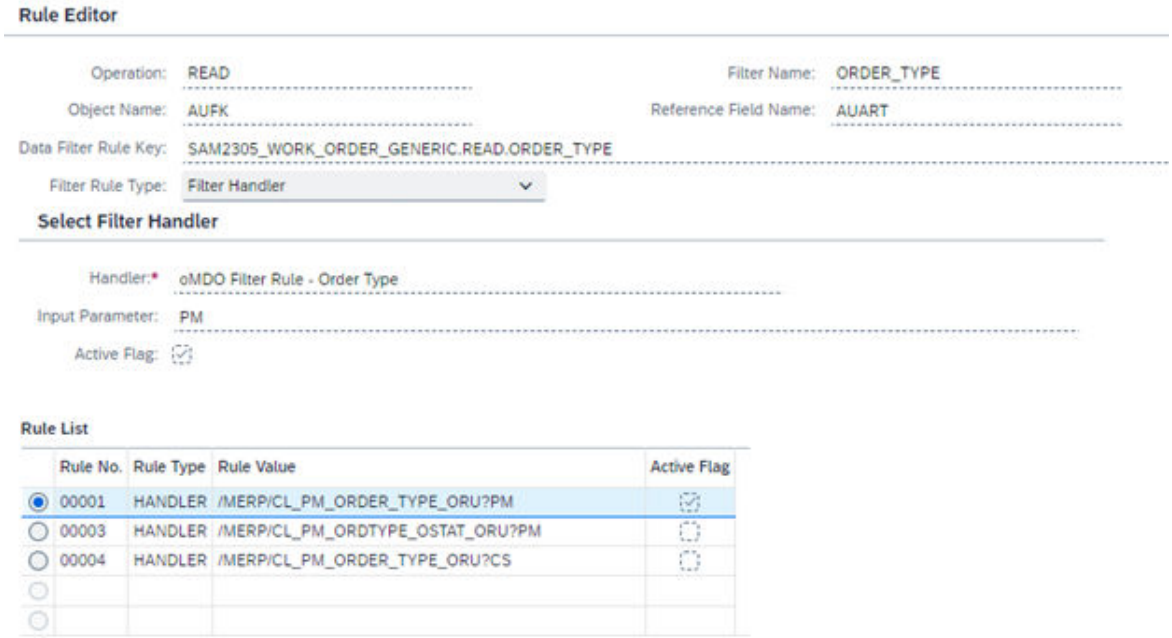
Procedure

1. Maintain assignment type filter *WO_ASSIGNMENT_TYPE* with only the following assignment types:



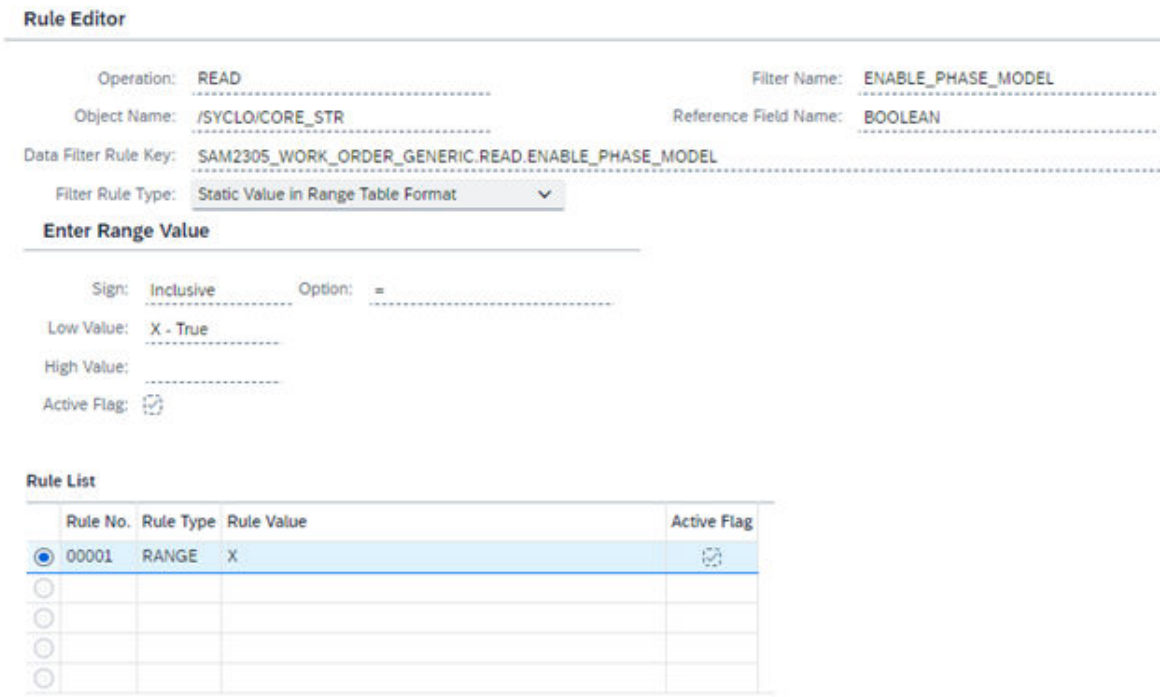
> ### Note
>
> Since the phase model could be configured for assignment type 2 and 6, the non-phase model also works for only those assignment types.

2. Set filter handler */MERP/CL_PM_ORDER_TYPE_ORU?PM* to active, which returns order types that have overall status profile assigned and those which are not.

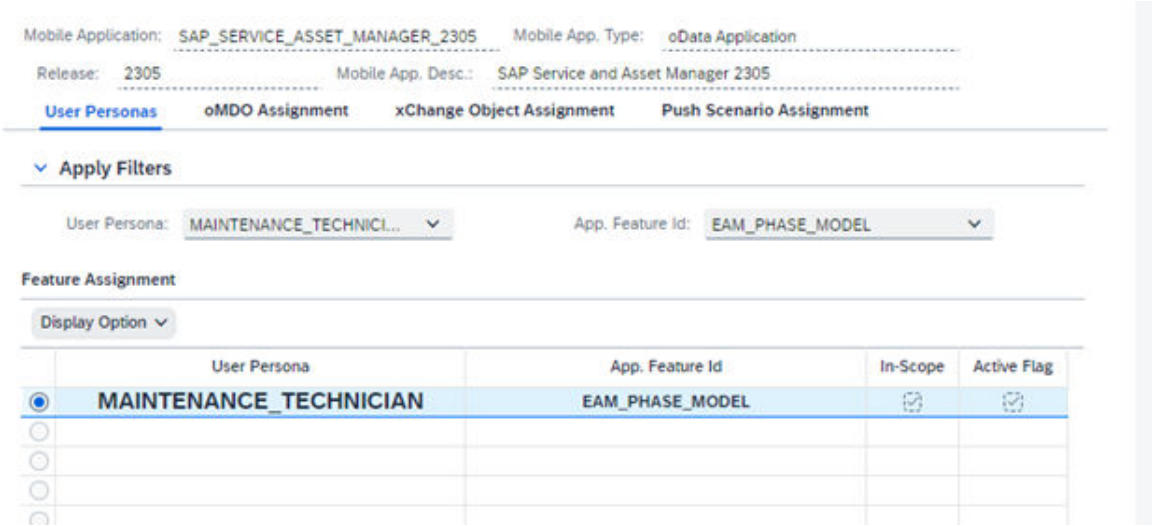


> **Note**
>
> As part of 2210 and previous release, this feature was disabled, however, it can now be active to fetch all PM order types, that is, phase model related and non-related.

3. Set *ENABLE_PHASE_MODEL* to active.

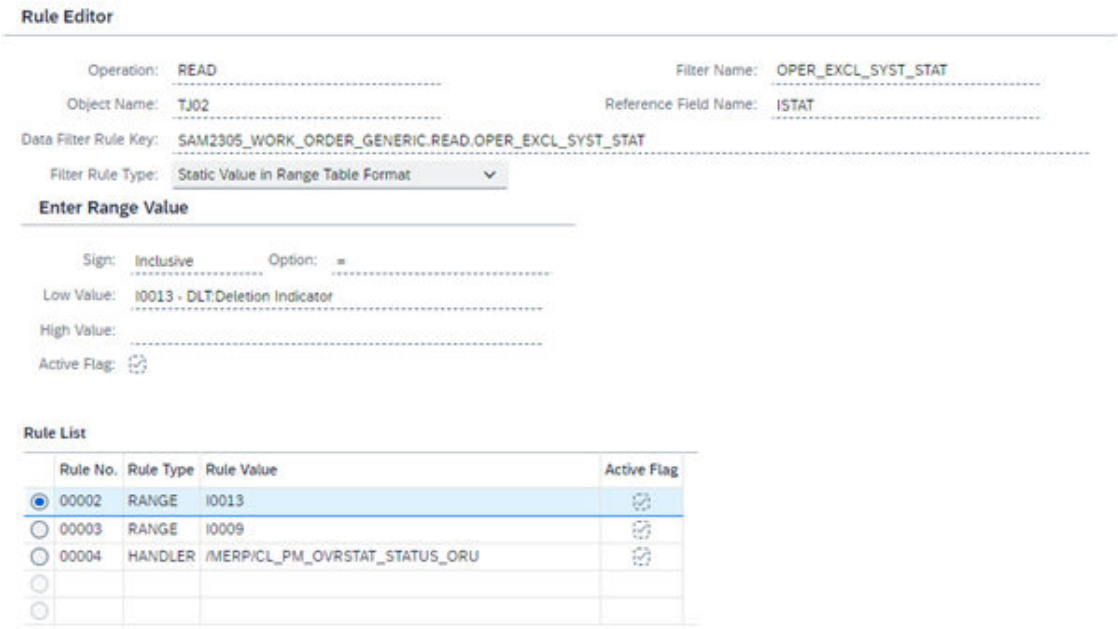


Activate the features in the Component assignments section of the configuration for the Maintenance Technician persona.



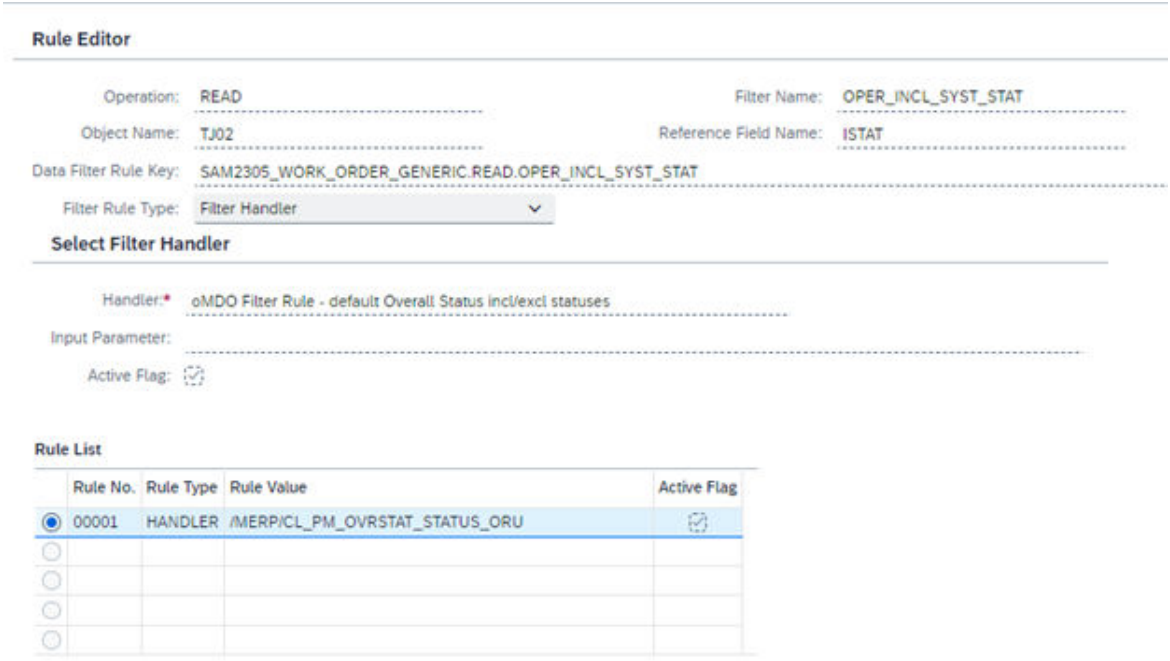
4. Make sure that filter handler */MERP/CL_PM_OVRSTAT_STATUS_ORU* is active, which by default injects system status `IEAM5` (ORSC - Ready for Scheduling) and `I0820` (JBFI – Job Finished) into *OPER_EXCL_SYST_STAT* filter.

> **Note**
>
> If the default statuses defined in filter handler don't fulfill your business scenario, you can manually add the system statuses individually to the filter.

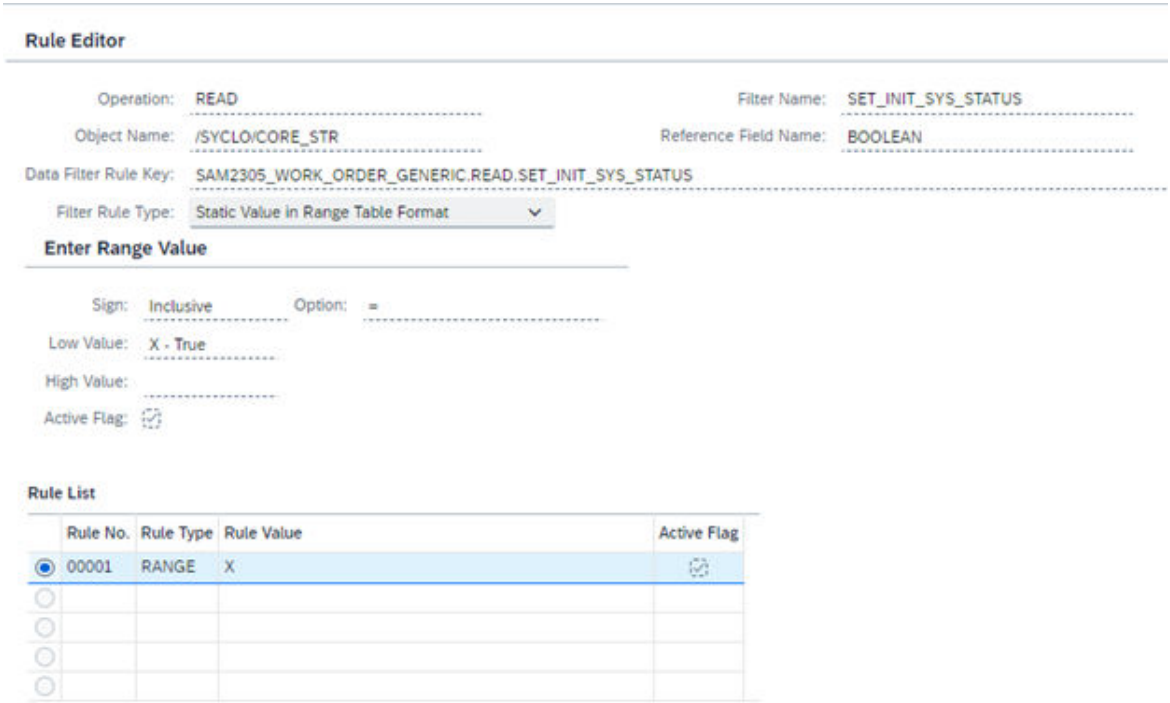


5. Make sure that the filter handler */MERP/CL_PM_OVRSTAT_STATUS_ORU* is active, which by default injects system status `I0117` (DSPT - Scheduled), `I0809` (JIPR – Job In Process), and `I0010` (PCNF – Partially Confirmed) into *OPER_INCL_SYST_STAT* filter.

> **Note**
>
> If the default statuses defined in filter handler don't fulfill your business scenario, you can manually add the system statuses individually to the filter.

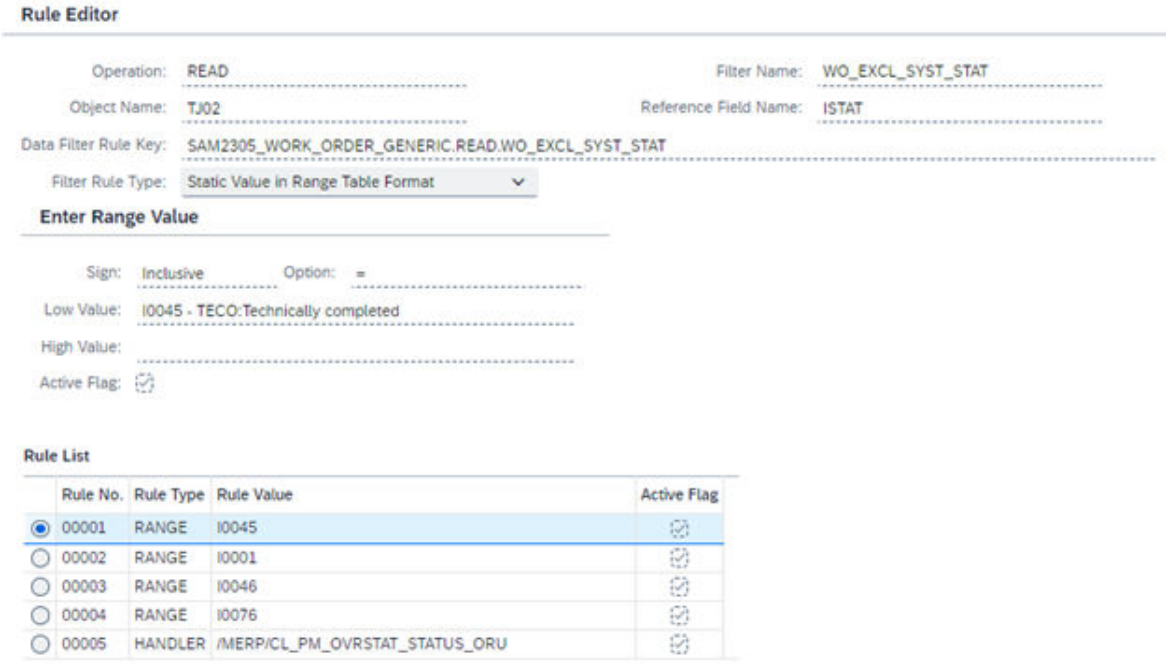


6. Set *SET_INIT_SYS_STATUS* filter to active. Initial mobile status will be determined for both overall status profile based and non-phase model relevant orders.



7. Set filter handler */MERP/CL_PM_OVRSTAT_STATUS_ORU* to active, which by default injects system status `IEAM6` (OMWC – Main Work Completed) and `I0820` (JBFI – Job Finished) into *WO_EXCL_SYST_STAT* filter.

   If the default statuses defined in filter handler don't fulfill your business scenario, you can manually add the system statuses individually to the filter.



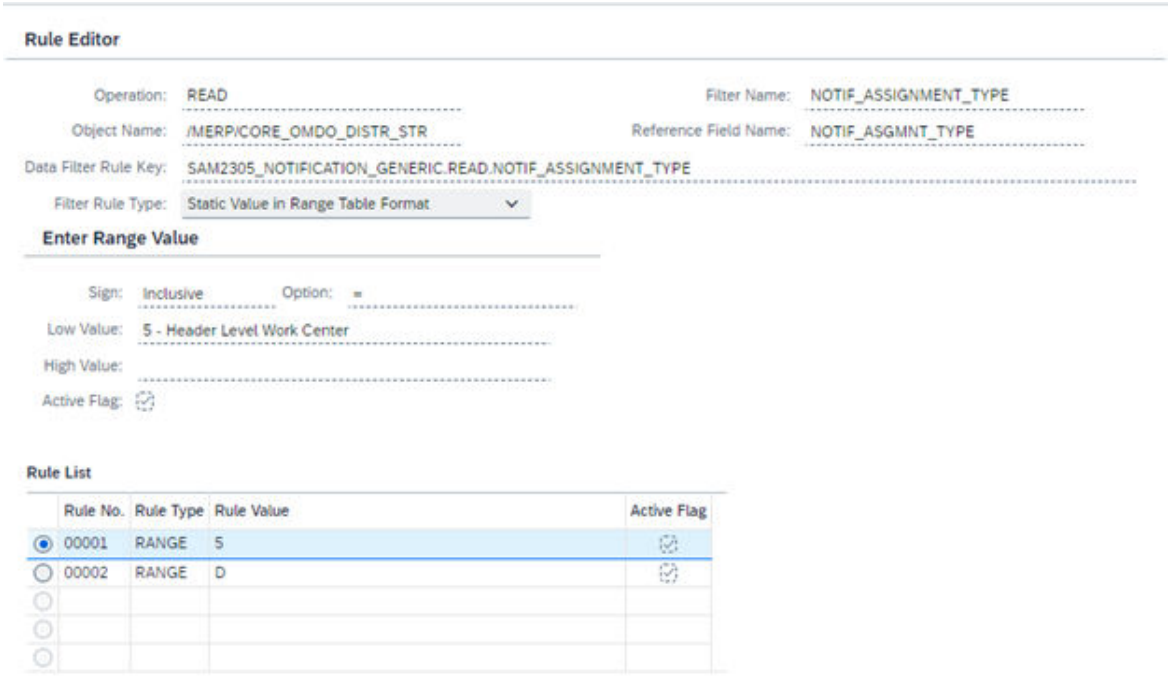
3.3.12.2 Notification Assignment Type Configuration

Configuring the Notification Assignment Type Filter

The following assignment types are supported:

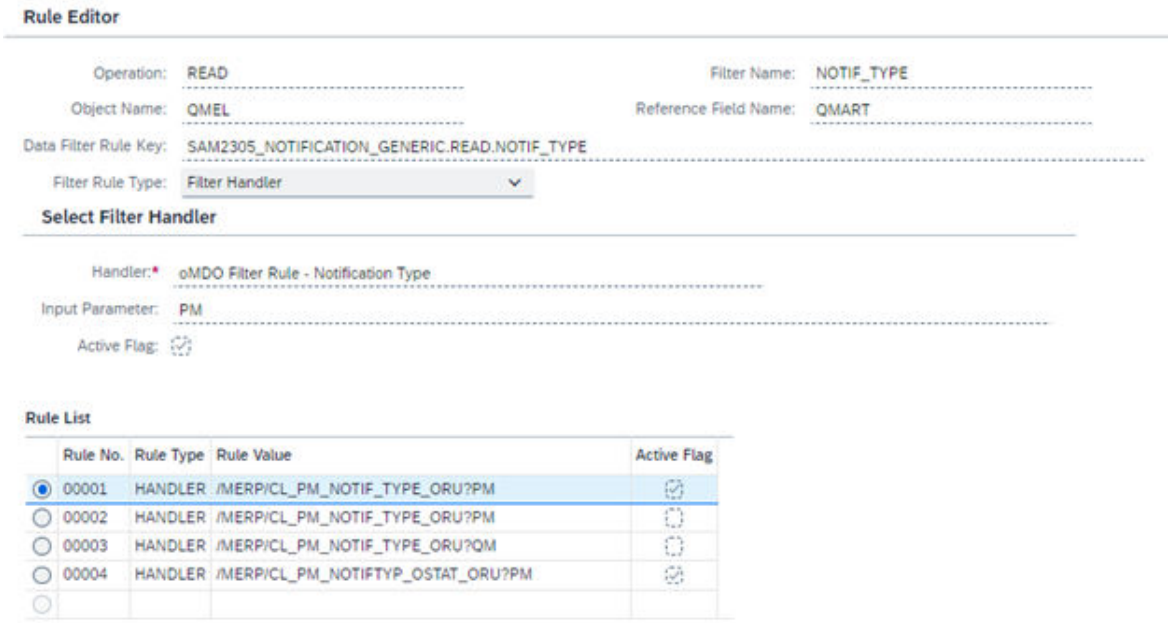
- 1: Header level person responsible
- 5: Header level work center
- D: Dependency queue

1. On the ConfigPanel home page, select *OData Mobile Data Object Configuration*. Make sure to select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. In the *OData Mobile Data Object List* select **SAM2310_NOTIFICATION_GENERIC**, and then the *Data Filter* tab.
3. Expand the *Defined Filters* list as follows; *Operation - READ* > *Data Distribution* > and click *NOTIF_ASSIGNMENT_TYPE*. Click the *Change* button.
4. Set *Low Value* with the desired assignment type as defined by the assignment type model.
5. *Save* your changes.



Configuring the Notification Type Filter

1. On the ConfigPanel home page, select *OData Mobile Data Object Configuration*. Make sure to select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. In the *OData Mobile Data Object List* select **SAM2310_NOTIFICATION_GENERIC**, and then the *Data Filter* tab.
3. Expand the *Defined Filters* list as follows; *Operation - READ* > *Data Distribution* > and click *NOTIF_TYPE*. Click the *Change* button.
4. Set the */MERP/CL_PM_NOTIFTYP_OSTAT_ORU?PM* filter handler to *Active*. This filter handler returns notification types that have the overall status profile assigned.
5. Keep the */MERP/CL_PM_NOTIF_TYPE_ORU?PM* and */MERP/CL_PM_NOTIF_TYPE_ORU?CS* filter handlers active. Notification types returned will be non-phase model relevant.
6. *Save* your changes.



3.3.12.3 Order Type oMDO Configuration

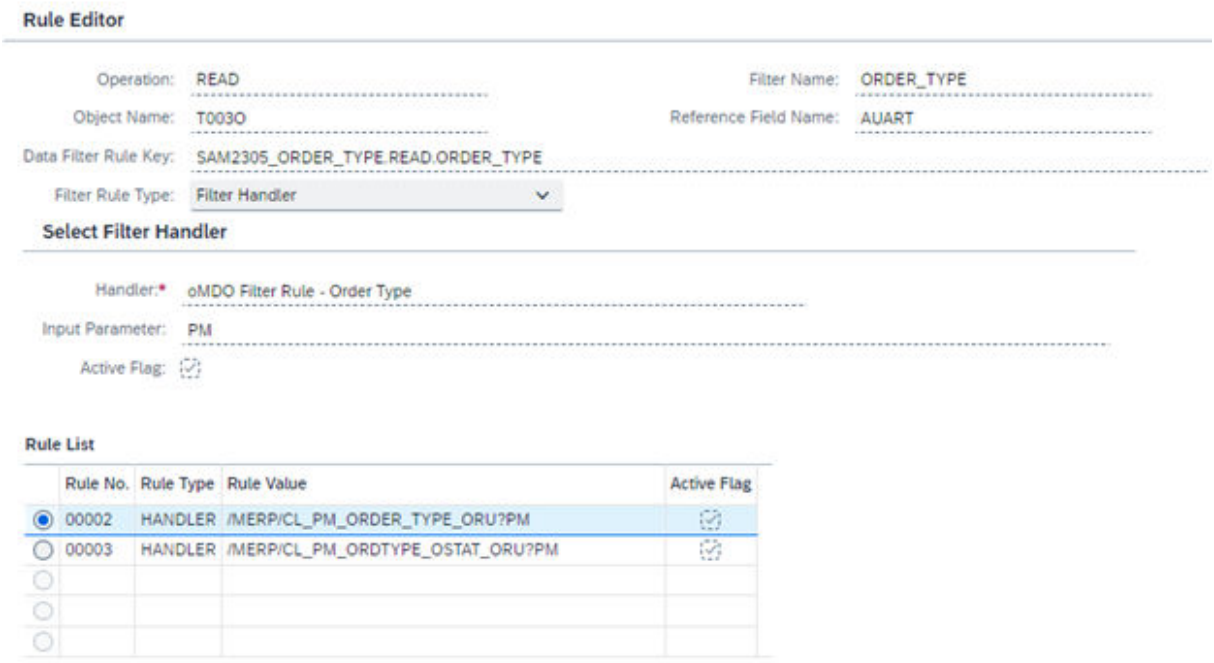
Context

Procedure

Set filter handler */MERP/CL_PM_ORDER_TYPE_ORU?PM* and/or */MERP/CL_PM_ORDER_TYPE_ORU?CS* to active as non-phase model relevant order types returned from the filter handler can be processed too.

/MERP/CL_PM_ORDER_TYPE_ORU?PM will return both phase and non-phase model relevant order types. The handler */MERP/CL_PM_ORDTYPE_OSTAT_ORU?PM* is redundant if both phase and non-phase model order types are to be processed.

> **Note**
>
> Set */MERP/CL_PM_ORDTYPE_OSTAT_ORU?PM* to active, which returns order types that have overall status profile assigned, if only the phase model relevant orders needs to be handled by app.



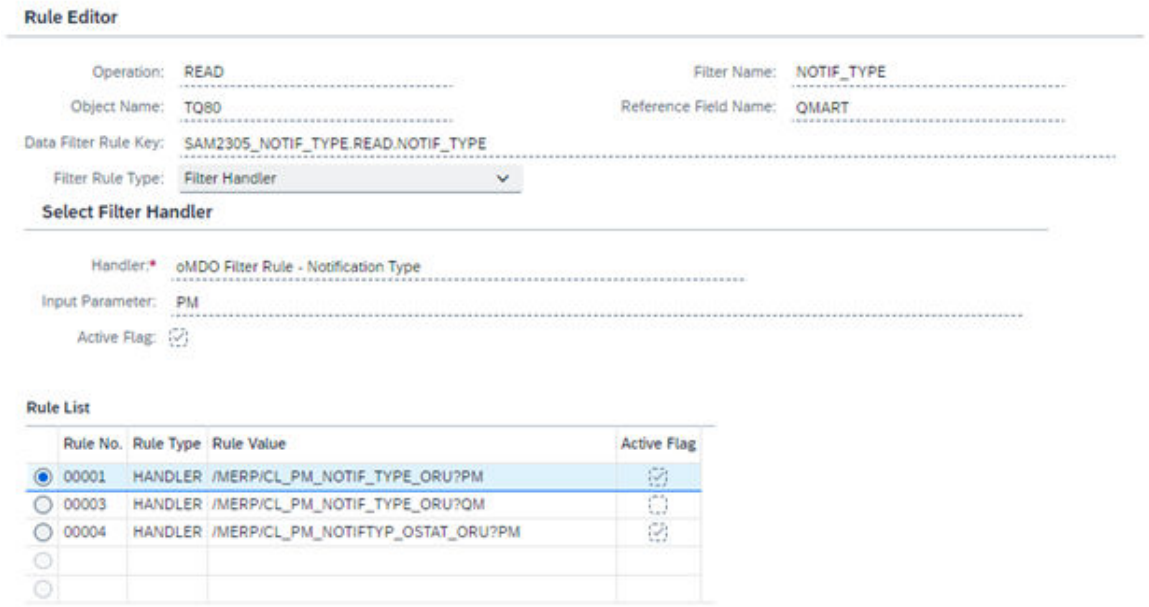
3.3.12.4 Notification Type oMDO Configuration

Context

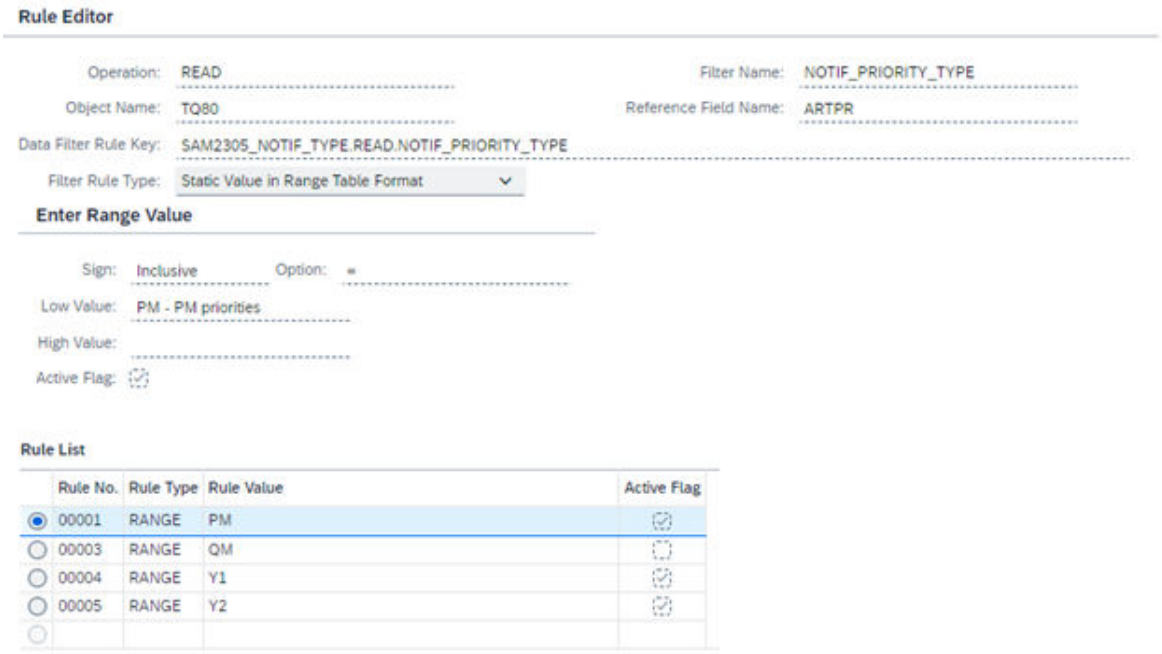
Procedure

1. Set filter handler */MERP/CL_PM_NOTIF_TYPE_ORU?PM* and/or */MERP/CL_PM_NOTIF_TYPE_ORU?CS* to active, which returns notification types that have overall status profile assigned and others.

   Filter handler */MERP/CL_PM_NOTIFTYP_OSTAT_ORU?PM* is redundant if both phase and non-phase relevant notification types are to be processed. Set filter handler *MERP/CL_PM_NOTIFTYP_OSTAT_ORU?PM* to active if only phase model relevant notification types need to be processed in the app.



2. Configure additional priority type values if notification uses priority type other than PM.



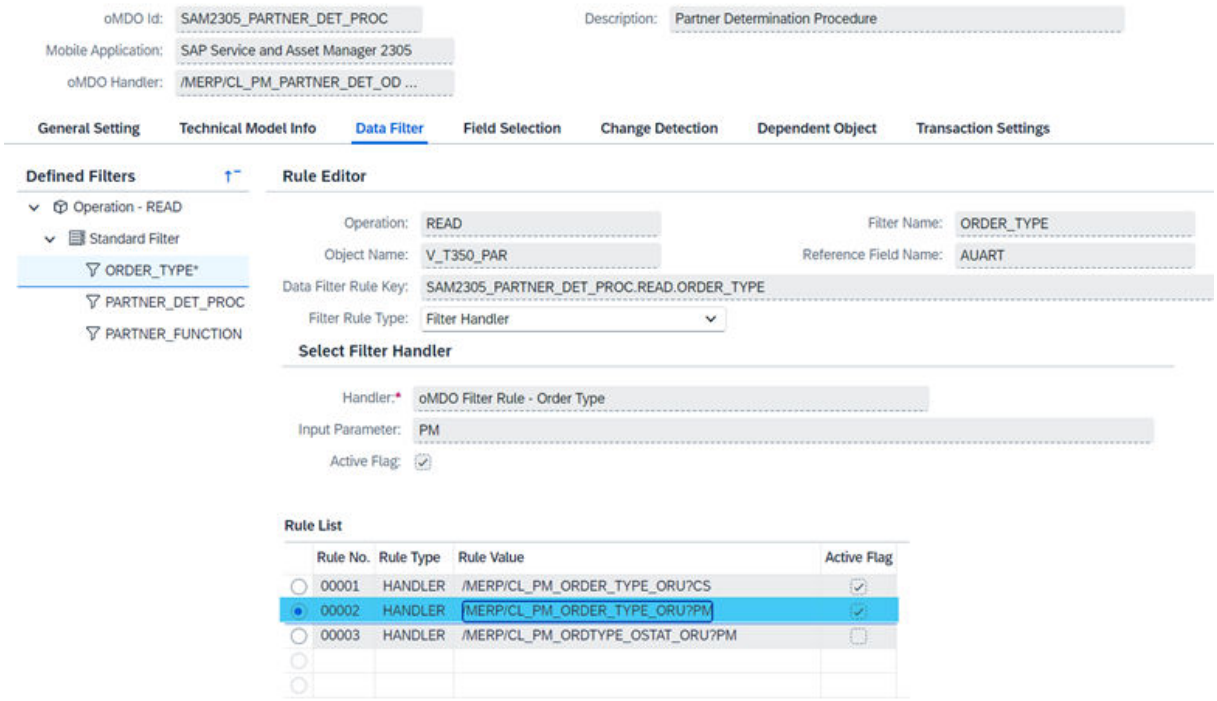
3.3.12.5 Work Order Partner Determination Procedure oMDO Configuration

Context

Procedure

The filter handlers */MERP/CL_PM_ORDER_TYPE_ORU?PM* and */MERP/CL_PM_ORDER_TYPE_ORU?CS* can stay active.

Filter handler */MERP/CL_PM_ORDTYPE_OSTAT_ORU?PM* can remain active, however it is redundant as it will pick up only the phase model enabled order types. Handler */MERP/CL_PM_ORDER_TYPE_ORU?PM* will take care of both phase model and non-phase model order types.

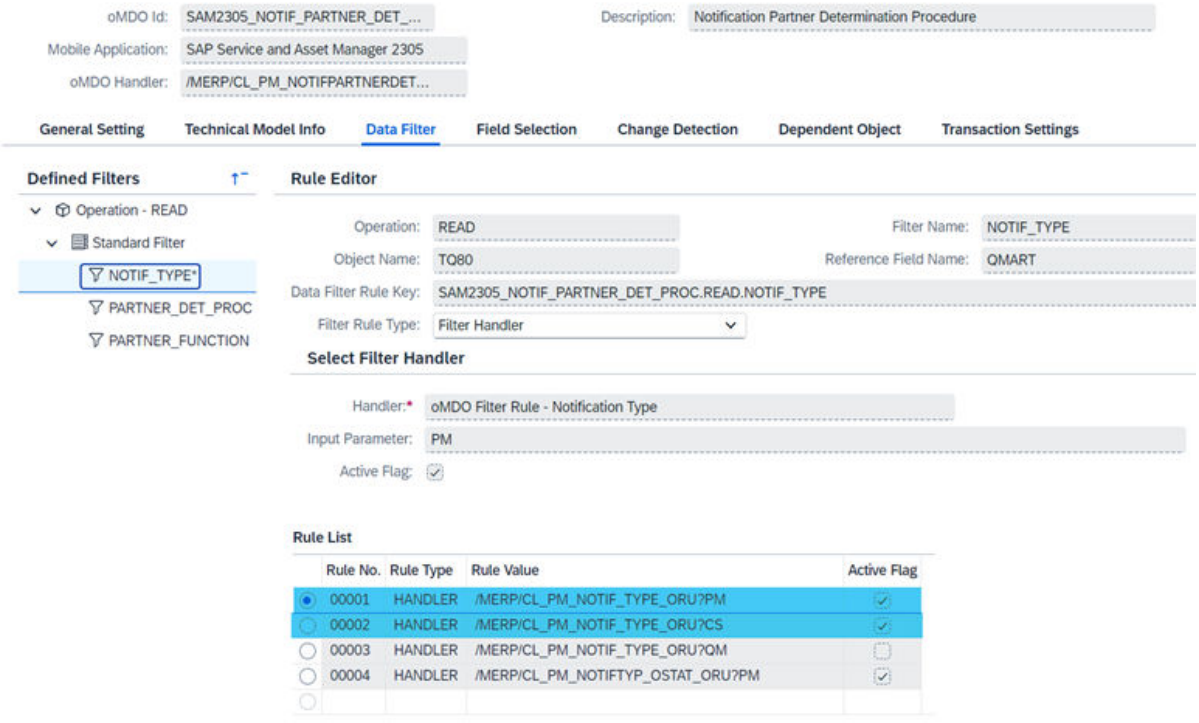


3.3.12.6 Notification Partner Determination oMDO Configuration

Context

Procedure

Set filter handler */MERP/CL_PM_NOTIF_TYPE_ORU?PM* (this will pull up both phase model and non-phase model notifications) and */MERP/CL_PM_NOTIF_TYPE_ORU?CS* to active. Handler */MERP/CL_PM_NOTIFTYP_OSTAT_ORU?PM* can remain active, however would be redundant if both phase and non-phase model notifications are pulled up.



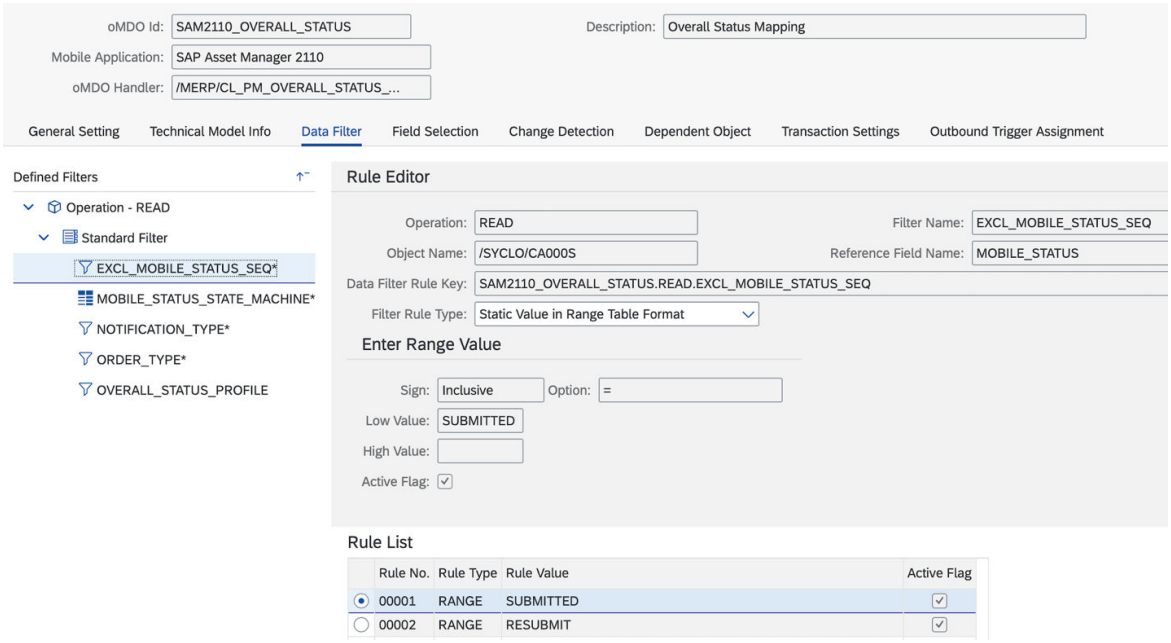
3.3.12.7 Overall Status oMDO Configuration

Context

Procedure

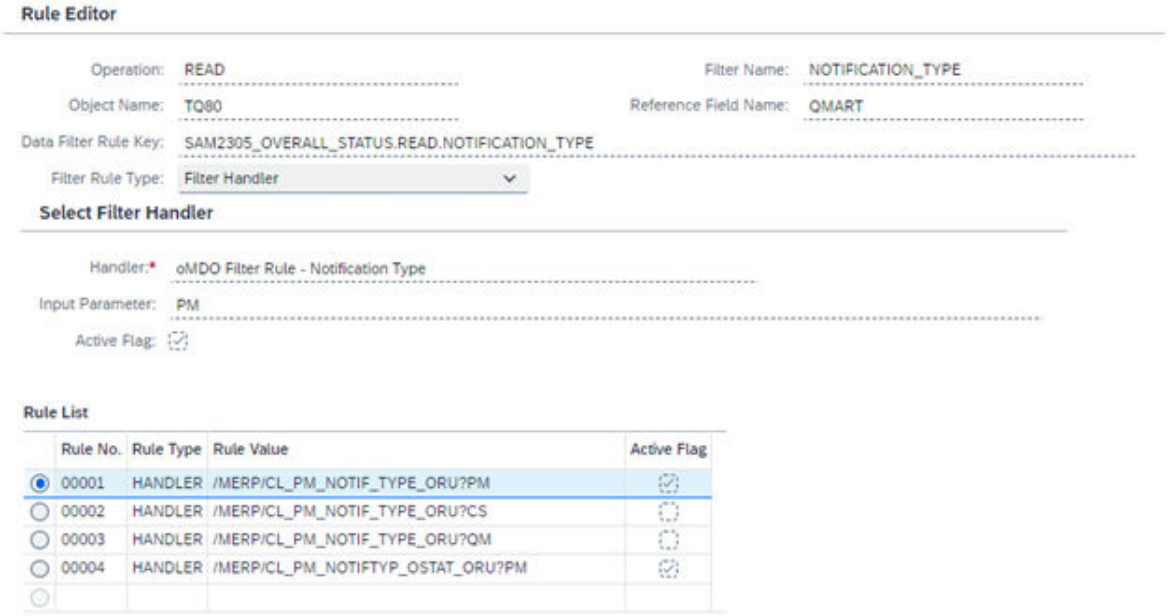
1. Use *EXCL_MOBILE_STATUS_SEQ* filter to disable certain phase model status transitions on your mobile client.

   The configuration of phase model status transition allows SUBMITTED to ACTIONREQ, but it isn't allowed in SAM, hence the following filter values must be active when phase model is enabled.



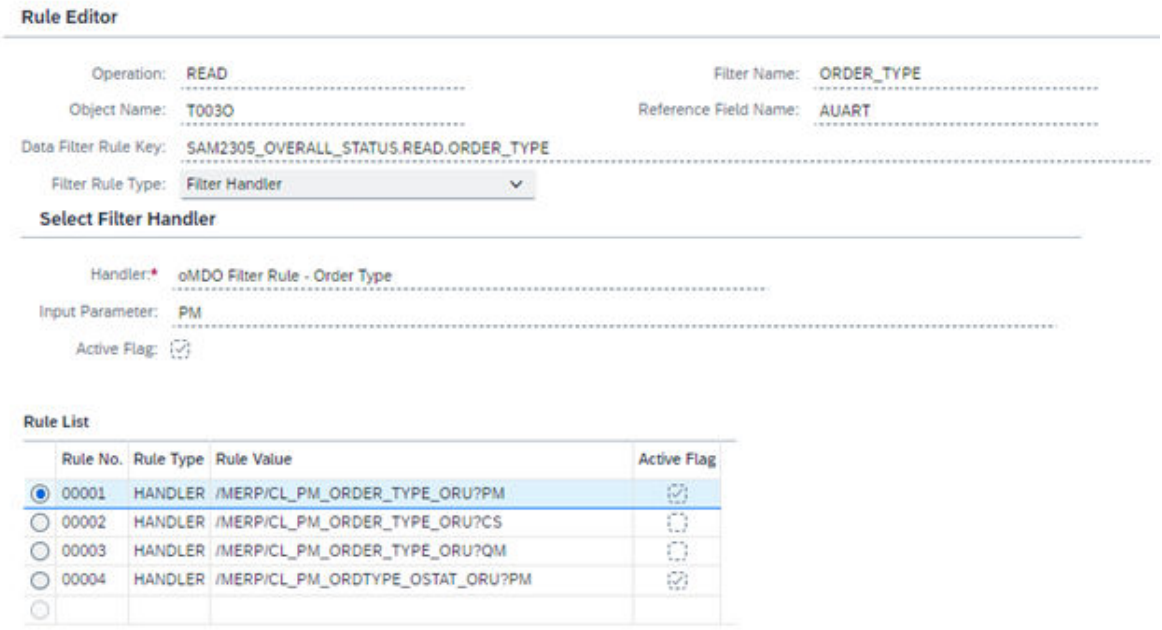
2. Make sure filter handler */MERP/CL_PM_NOTIF_TYPE_ORU?PM* is active, which returns notification types that have overall status profile assigned as well as non-phase model relevant ones.

   Filter */MERP/CL_PM_NOTIFTYP_OSTAT_ORU?PM* is redundant if both phase and non-phase notifications are to be processed in the app. Set filter handler */MERP/CL_PM_NOTIFTYP_OSTAT_ORU?PM* to active if only phase model relevant notification types need to be processed in the app.



3. Make sure filter handler */MERP/CL_PM_ORDER_TYPE_ORU?PM* is active, which returns order types that have overall status profile assigned as well as non-phase model relevant ones.

   Filter */MERP/CL_PM_ORDTYPE_OSTAT_ORU?PM* is redundant if both phase and non-phase orders are to be processed in the app. Set filter handler */MERP/CL_PM_ORDTYPE_OSTAT_ORU?PM* to active if only phase model relevant order types need to be processed in the app.



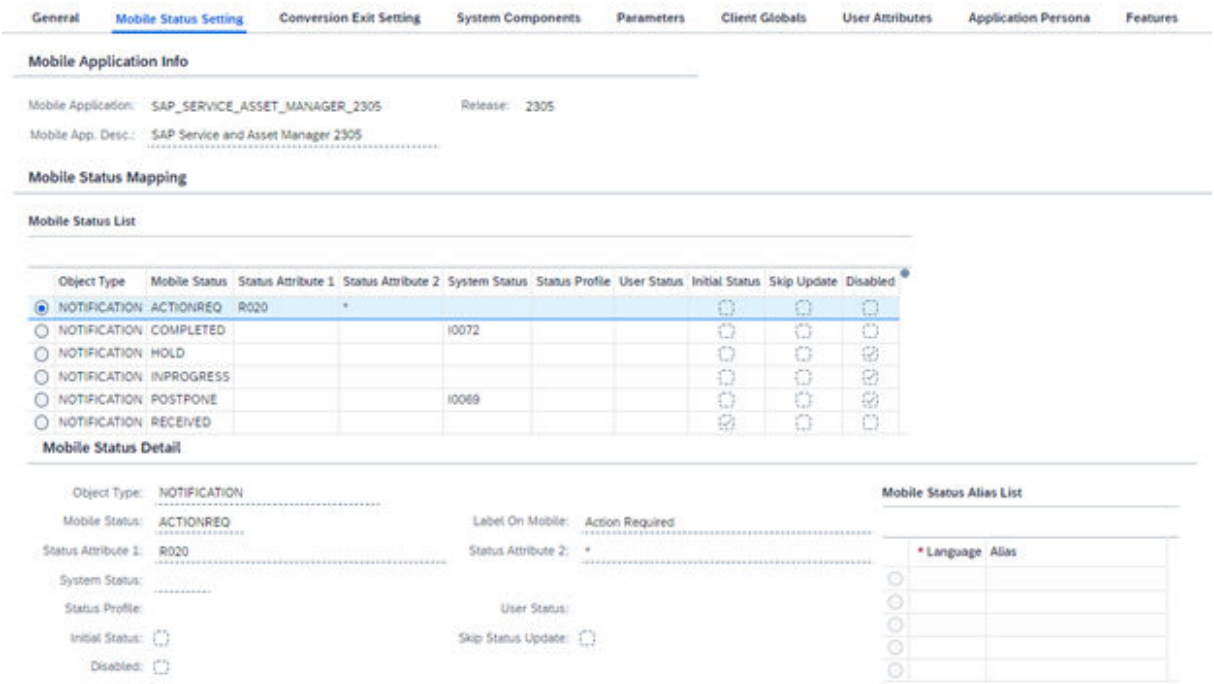
3.3.12.8 Mobile Status Mapping Configuration

The ConfigPanel shows you which PM mobile statuses are mapped to phase model statuses.

From the ConfigPanel home page, navigate to *Mobile Application Configuration* > *Mobile Status Setting* tab. Select your application in the *Defined Mobile Applications* list. Click the *Change* button to change the configuration.

The *Status Attribute 1* and *Status Attribute 2* fields are used for the status mapping as follows:

- Status Attribute 1: Overall status
- Status Attribute 2: Overall status profile. If an asterisk (*) is present, it means any overall status profile.



If you're enabling the phase model, activate the following mobile statuses by setting the *Status Attribute 1* and *Status Attribute 2* fields as indicated in the table. In an out of the box installation, the statuses are inactive, as the phase model is disabled by default.

> ⓘ Note
>
> Set any other statuses not indicated in the following to *Disabled* when you enable the phase model.

Mobile Status Mapping

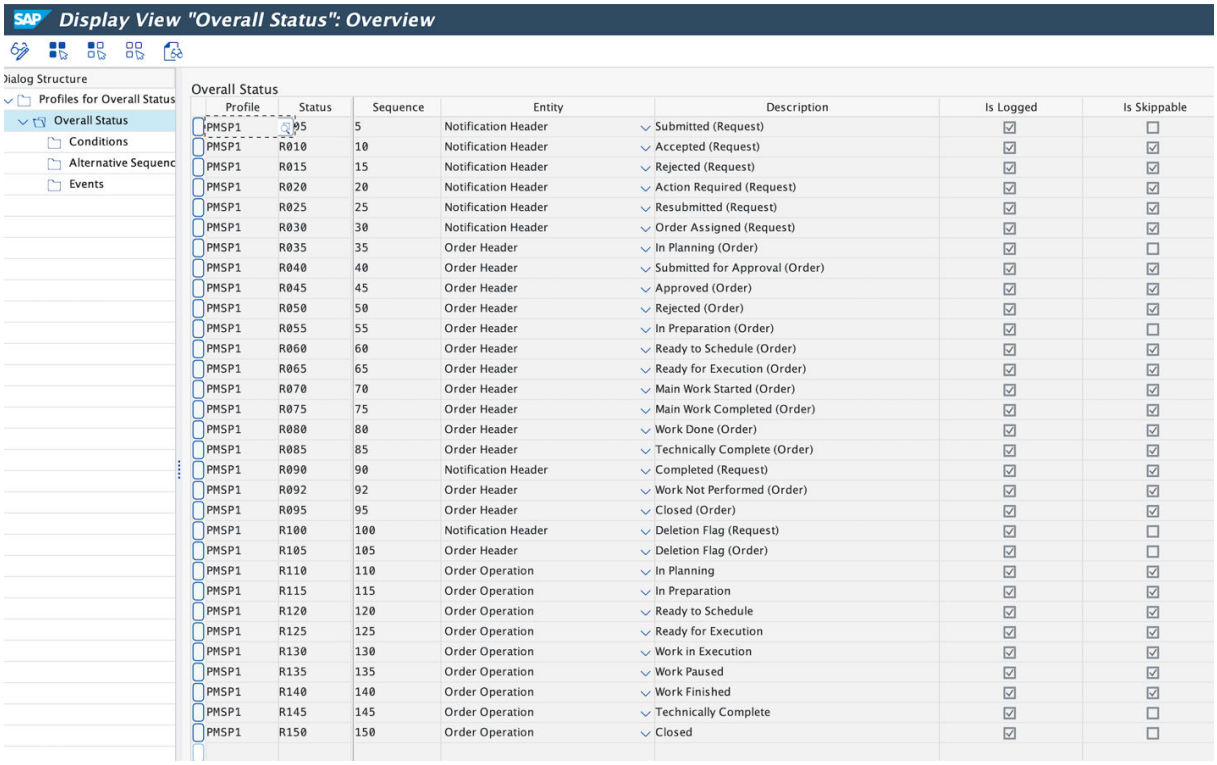
Object Type	Mobile Status	Status Attribute 1	Status Attribute 2	System Status
NOTIFICATION	ACTIONREQ	R020	*	

Object Type	Mobile Status	Status Attribute 1	Status Attribute 2	System Status
NOTIFICATION	COMPLETED Status is used in both phase model ON and OFF scenarios			I0072
NOTIFICATION	STARTED	R025	*	I0070
NOTIFICATION	STARTED Status is used in both phase model ON and OFF scenarios			
NOTIFICATION	SUBMITTED	R005	*	
WORKORDER	COMPLETED	R075	*	
WORKORDER	DONE	R080	*	
WORKORDER	READY	R065	*	
WORKORDER	STARTED	R070	*	
WO_OPERATION	COMPLETED	R140	*	
WO_OPERATION	HOLD	R135	*	
WO_OPERATION	READY	R125	*	
WO_OPERATION	STARTED	R130	*	

Save any changes you've made.

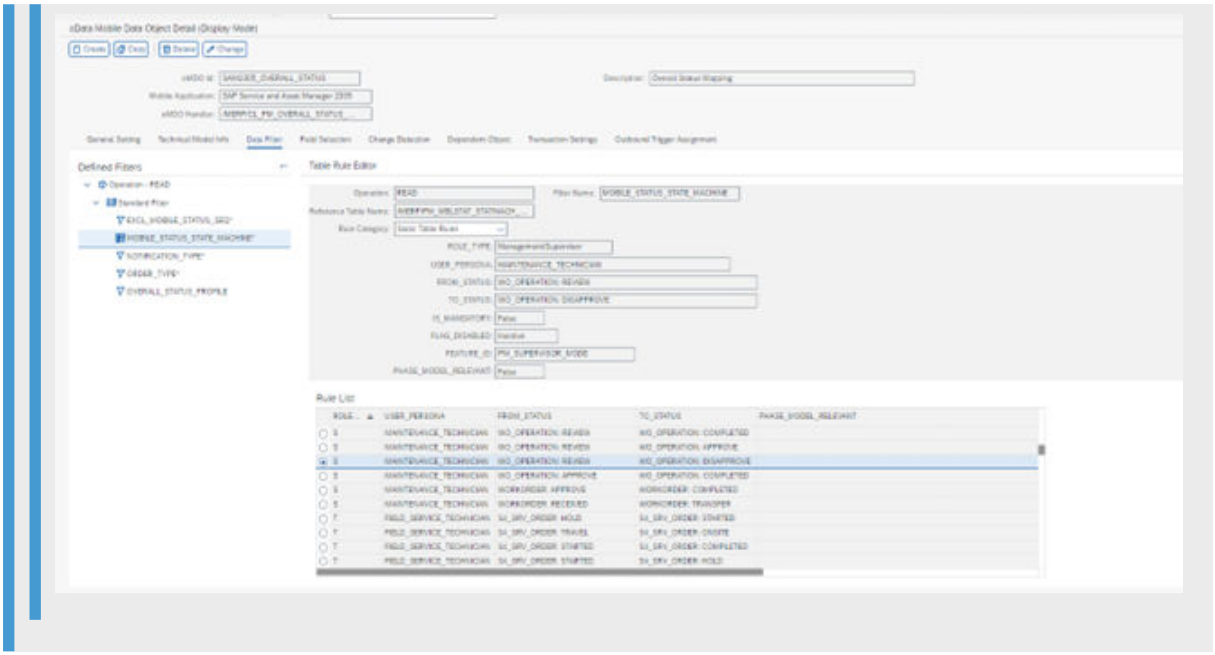
3.3.12.9 Mobile Status State Machine

The status transition of phase model is maintained in *IMG* > *Plant Maintenance* and *Customer Service* > *Maintenance* > *Service Processing* > *Fiori Apps* for *Maintenance Processing* > *Configure Overall Status Profiles*.



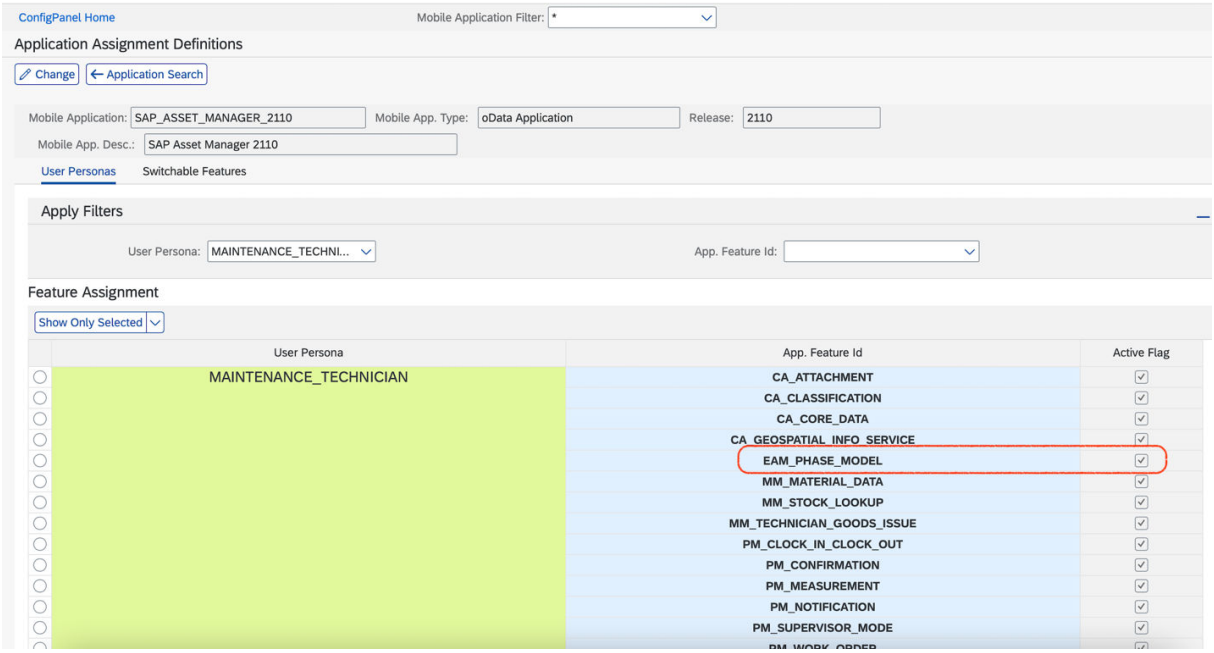
Status transition information is downloaded as *EAMOverallStatusSeqs* entity set to the mobile client, which determines status transition on the device.

> ### Note
>
> As of the SAP Service and Asset Manager 2305 release, SAP is adding new state transitions to the mobile state machine for the supervisor role and the technician role within the maintenance technician persona.
>
> Configure the state transitions (add, remove, activate, or disable) in the state machine for the corresponding phase model flag. The data filter *MOBILE_STATUS_STATE_MACHINE* is found in *SAM2305_OVERALL_STATUS*.
>
> The state transitions corresponding to the phase model in the state machine are added based on the phase model state profile from the core configuration. The state machine is the driving factor and NOT the state transition in IMG (*IMG* > *Plant Maintenance and Customer Service* > *Maintenance Service Processing* > *Fiori Apps for Maintenance Processing* > *Configure Overall Status Profiles*) as of SSAM2305.
>
> > ### Note
> >
> > New fields USER_PERSONA, FEATURE_ID and PHASE_MODEL_RELEVANT have been added to the state machine table to indicate transitions specific to persona, feature and if they are relevant to the phase model only. For example, the highlighted status transition on the screenshot below allows a supervisor with a maintenance technician persona to not approve a work order operation in review status for an order type without a phase model when the PM_SUPERVISOR_PHASE model feature is enabled.

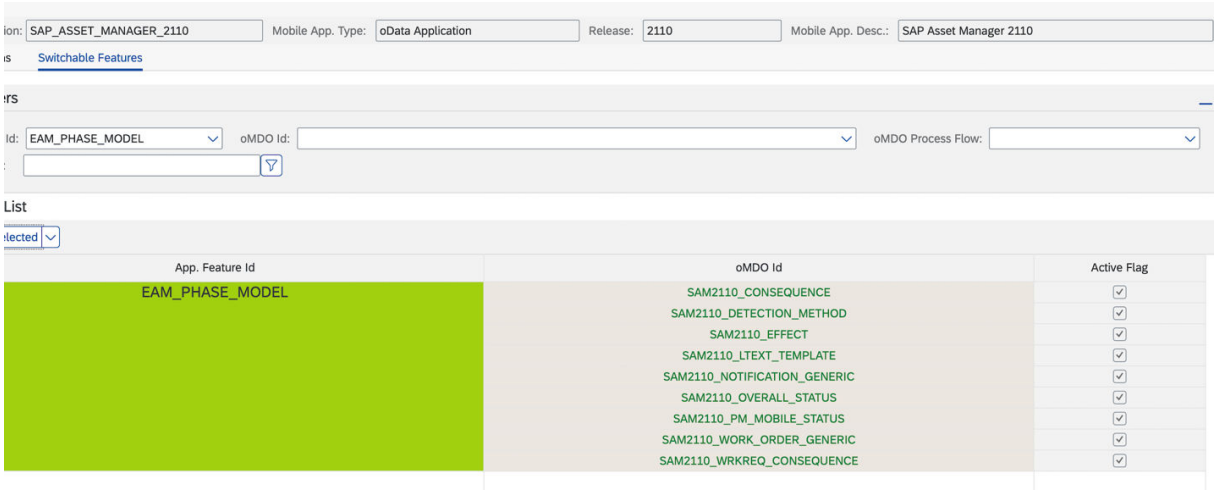


3.3.12.10 Persona and Feature Enablement

EAM phase model is added as persona and feature assignment. When phase model is enabled, the following persona/feature assignment must be enabled. By default, the following setting is inactive:



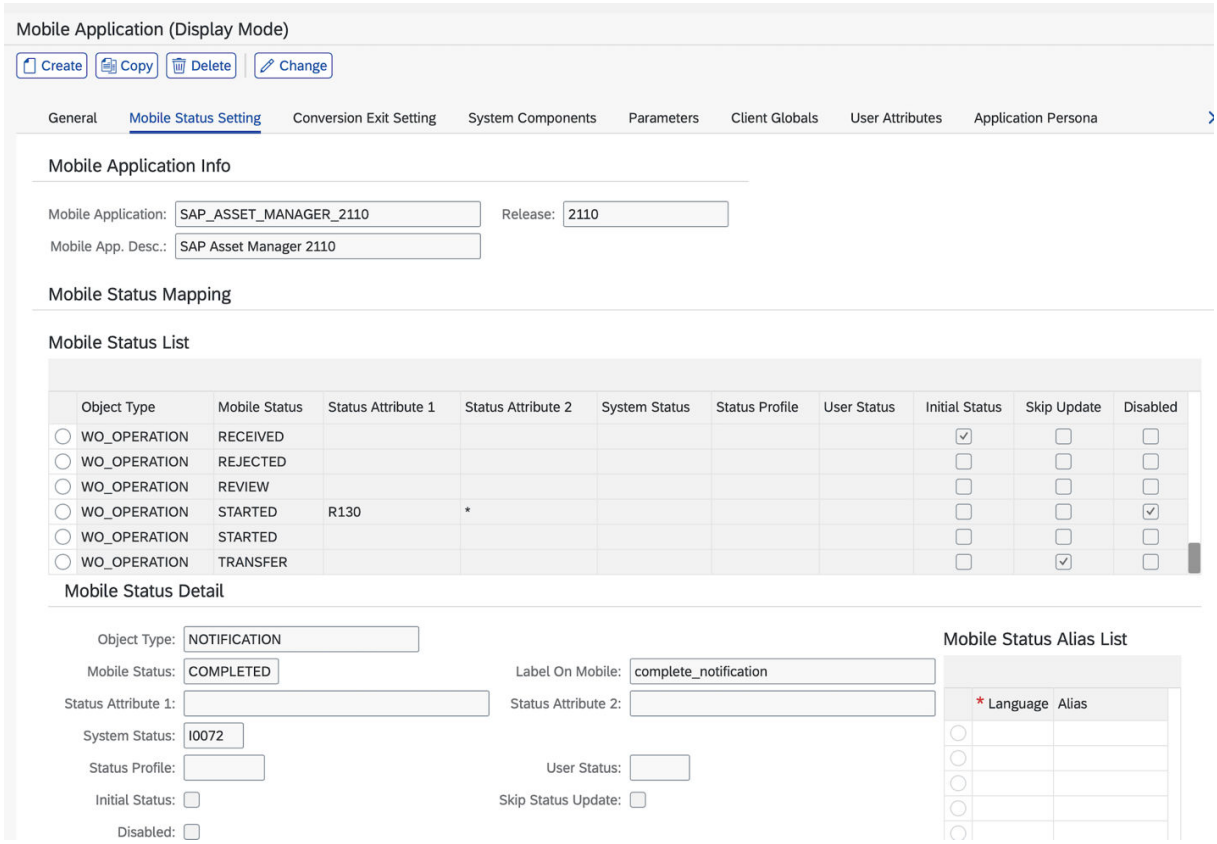
When *EAM_PHASE_MODEL* feature is active, phase model-related data is downloaded during synchronization, for example *WorkRequestConsequence* or *Effect*.



When phase model is disabled, mobile status in SAM 2110 behaves the same as in previous versions. However, status state machine is added to SAM 2110. The related configurations are the following:

Mobile Status Mapping Configuration

Mobile status mapping is still maintained in Configuration Panel with phase model-related statuses disabled in SAM 2110.



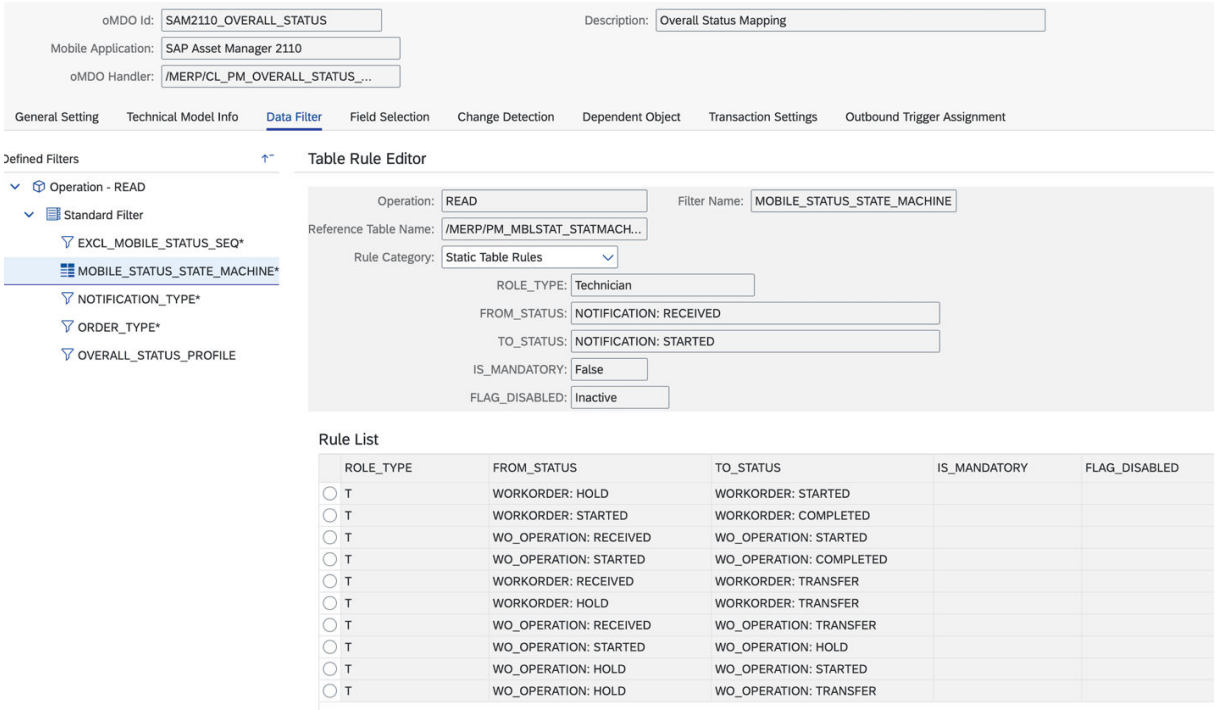
The following mobile status mapping must be active when phase model is disabled:

Mobile Status Mapping Configuration

Object Type	Mobile Status	Status Attribute 1	Status Attribute 2	System Status
NOTIFICATION	COMPLETED *this status is used in both phase model ON and OFF scenarios.			I0072
NOTIFICATION	POSTPONE			I0069
NOTIFICATION	RECEIVED			
NOTIFICATION	STARTED *this status is used in both phase model ON and OFF scenarios.			I0070
TASK	COMPLETED			I0156
TASK	RECEIVED			
TASK	STARTED			I0155
TASK	SUCCESS			I0157
WORKORDER	COMPLETED			I0045
WORKORDER	HOLD			
WORKORDER	RECEIVED			I0630
WORKORDER	REJECTED			
WORKORDER	REVIEW			
WORKORDER	STARTED			I0002
WORKORDER	TRANSFER			
WO_OPERATION	COMPLETED			
WO_OPERATION	HOLD			
WO_OPERATION	RECEIVED			
WO_OPERATION	REJECTED			
WO_OPERATION	REVIEW			
WO_OPERATION	STARTED			
WO_OPERATION	TRANSFER			

Mobile Status State Machine Configuration

Mobile status state machine is added as filter in Overall Status oMDO (SAM2110_OVERALL_STATUS). This configuration allows us to configure all status transition on mobile client, given the mobile status is already maintained in the configuration of mobile status mapping.



ROLE_TYPE property allows you to define status transition record to be relevant for Technician or Supervisor. Any supervisor-specific status records are used only if Supervisor mode is enabled.

When phase model is enabled, this filter would simply be ignored.

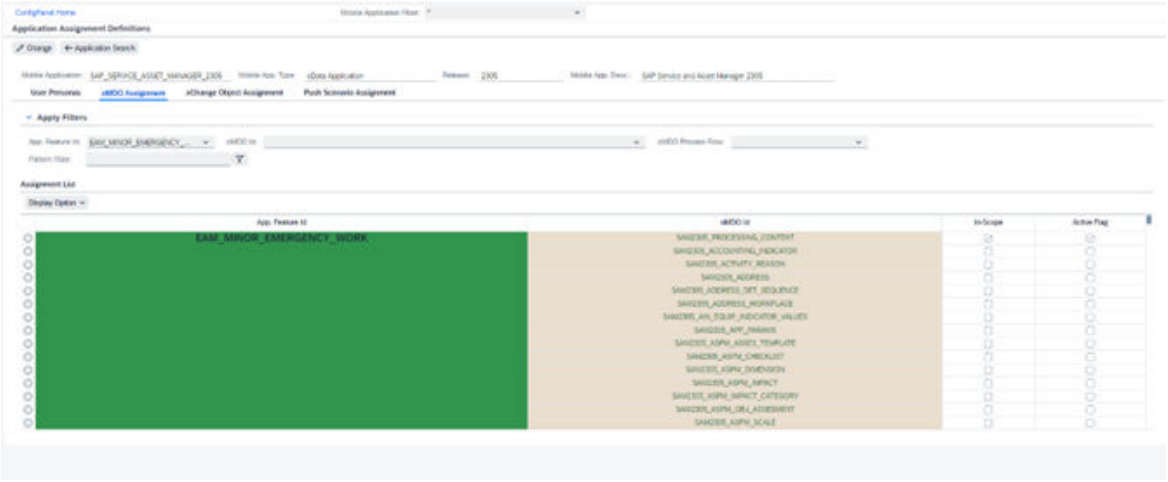
3.3.12.11 Minor Emergency Work

This new feature is added to support emergency or minor work in a reactive maintenance phase model. It helps to skip phases and issue a work order or notification when a technician is authorized to create those items.

If the technician is authorized, he will see an option called *Minor Work*. When this option is selected and the request is submitted, the request is not sent to the screening technician for approval, but is automatically approved and is moved to the planning phase.

As for 2305 release:

- A new *EAM_MINOR_EMERGENCY_WORK* feaure is created and assigned to a new *NotificationProcessingContexts* entity linked to the *NotifProcessingContext* property in the *MyNotificationHeaders* entity
- A new *OrderProcessingContext* field is added to the *MyWorkOrderHeader* entity. It is intended to be used as a filter for reading WOs and operations.
- The user must have authority (the *I_EXCP_MN* object: *Exception Process* > *Maintenance Notification*) to be able to create a notification with an exception process.



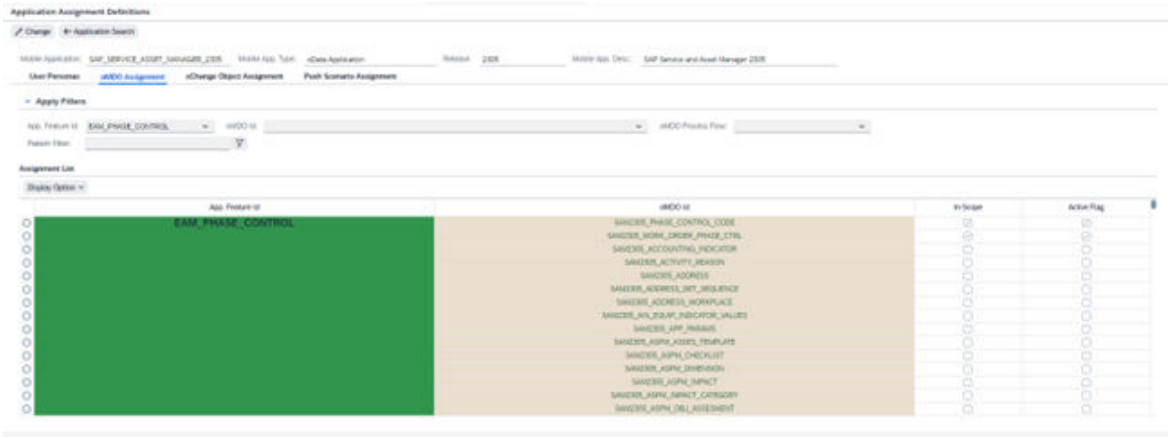
> **Note**
>
> This feature is available in S/42022, that is, in S4CORE 107 and beyond, as a basic feature.

3.3.12.12 EAM Phase Control Code

The process control code can limit a certain process state change based on the system state. It can also be applied to the execution phase, which prevents the user from moving the order or operation from one sub-phase to another.

As for 2305 release:

- IMG backend configuration path is the following: *IMG* > *Logistics* > *General* > *Product Lifecycle Management (PLM)* > *Plant Maintenance and Customer Service* > *Maintenance and Service Processing* > *Fiori Apps for Maintenance Processing* > *General Settings* > *Define Phase Control Code for Maintenance Orders*
- The new feature *EAM_PHASE_CONTROL* is created to enable the EAM phase control code. To block the transition of an order or operation from one phase to another when the phase model function is enabled. If the EAM_PHASE_CONTROL feature is active, the following entities are enabled.
  - PhaseControl
  - PhaseControlCode
  - PhaseControlKey
  - PhaseControlSystemMobileStatusMap
  - PhaseControlSystemStatus
  - WorkOrderOperationPhaseControl
  - WorkOrderPhaseControl
- This feature is active right out of the box. To disable it, refer to the feature configuration.



> **Note**
>
> For this feature to work, the *EAM_PHASE_MODEL* feature must be active in the SSAM application configuration.

> **Note**
>
> This feature is available for the S4CORE 106 and subsequent versions.

3.4 Geospatial Service Definitions

Geospatial Services provide the technology to create, analyze, maintain, and distribute geospatial data and information.

You can use either GIS (Geographic Information System) or GEF (Geographical Enablement Framework) with the app. Note that you can use either service, but you can't use both Services at the same time in the application.

> **Note**
>
> Esri base maps are the only base maps supported for the app.

3.4.1 Geographic Information System (GIS)

A geographic information system (GIS) integrates hardware, software, and data for capturing, managing, analyzing, and displaying all forms of geographically referenced information.

The SAP Service and Asset Manager application has custom map controls with GIS functionality implemented using Mobile Development Kit and Open UI extensions. The SAP Service and Asset Manager application is delivered with predefined globals that are Esri-specific, however, you can point to any GIS vendor you choose.

For information about configuring GEF, see the subtopics located Geographical Enablement Framework (GEF) [page 206] topic.

3.4.1.1 Configuring Your Map Settings

Some GIS settings are standard with the initial SAP Service and Asset Manager application.

You can change any of the settings described in this topic to configure the application for your site.

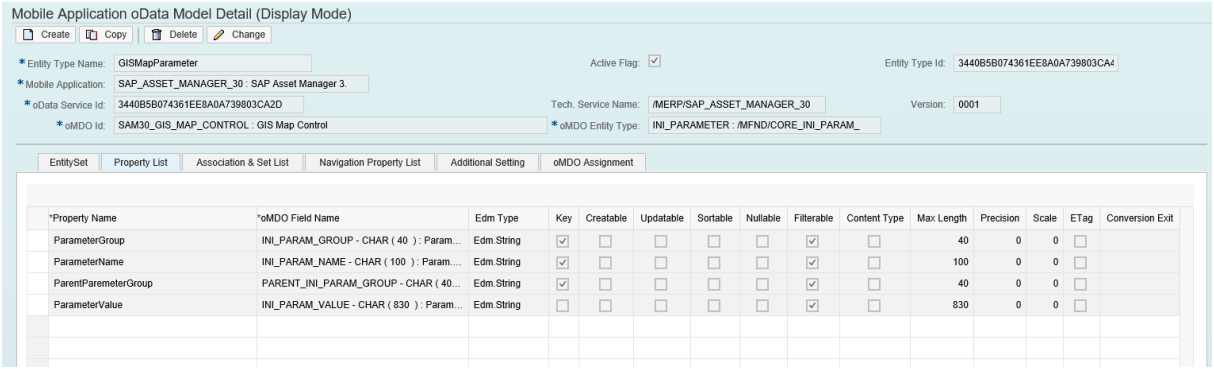
> **Note**
>
> You can also change the map setting metadata through the Mobile Development Kit. Note that if there are metadata differences, Mobile Development Kit changes override ConfigPanel changes.

OData Model Configuration - Property List Tab

In the ConfigPanel, the *GISMapParameter* entity type contains the following properties:

- ParameterGroup
- ParameterName
- ParentParameterGroup
- ParameterValue

Use the fields in the following section to properly categorize these parameters.



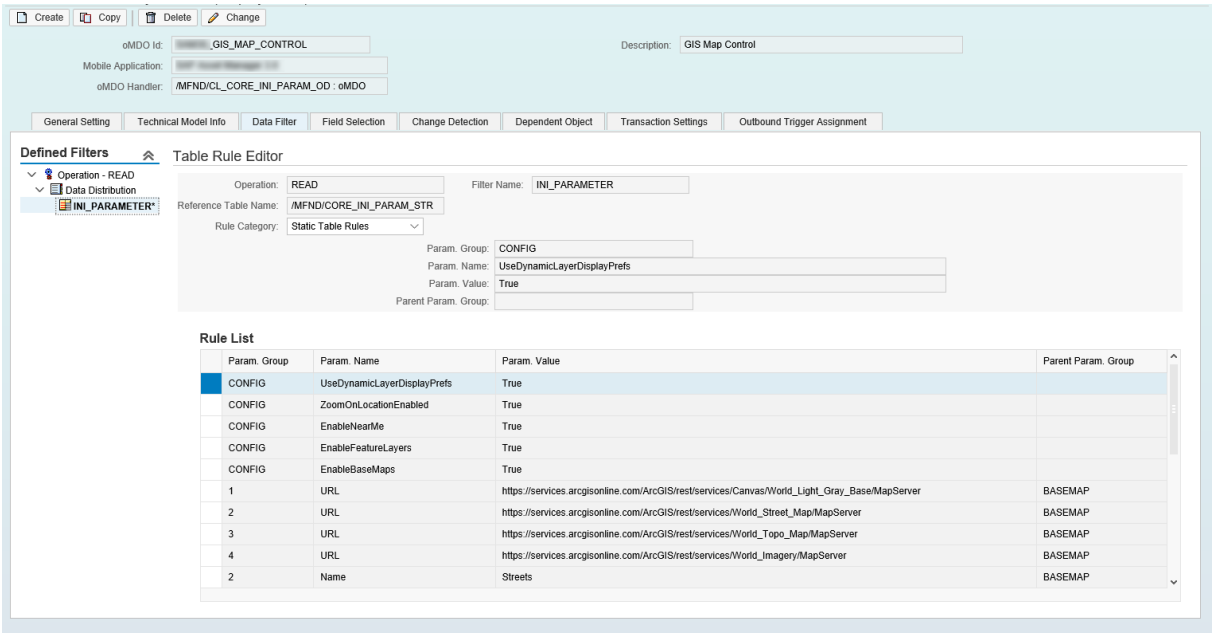
OData Mobile Data Object Configuration - Data Filter Tab

From the ConfigPanel Home page, navigate to *OData Mobile Data Object Configuration* > *Data Filter Tab* > *<APP_VERSION>_GIS_MAP_CONTROL* > *Operation - READ* > *Data Distribution*. Click the *Change* button.

The following parameters are standard:

- **CONFIG:** Parameter Group
- **BASEMAP:** Parent Parameter Group
- **FEATURELAYER:** Parent Parameter Group

See the following for an example screenshot of the parameters in the ConfigPanel in the *Data Filter* tab, and a table representing how to configure the parameters in the tab.



Parameter Group	Parameter Name	Parameter Value	Parent Parameter Group	Description
1	URL	<https://>	BASEMAP	Basemap URL
2	Name	Text value	BASEMAP	Basemap display name
1	Filename	<filename.tpk>	BASEMAP	Offline filename for basemap
1	Name	Text value	FEATURELAYER	Display name of feature layer
1	URL	<https://>	FEATURELAYER	Feature layer URL
1	Properties	**Example** { "ID": "{objectid}", "Title": "{metamorphic_facies}", "HeadlineText": "{geomodifications}", "Subhead": "{comments}" }	FEATURELAYER	Feature layer properties/ attributes to display in object cell. Map properties to `ObjectCell`fields
1	Actions	[]	FEATURELAYER	Array of actions to run for this feature layer.

Parameter Group	Parameter Name	Parameter Value	Parent Parameter Group	Description
CONFIG	`EsriClientID`	<Enter Esri License Key>		Esri runtime license string. **Note** For more information about how to get license to ESRI ClientID, see ArcGIS Developers documentation.
CONFIG	`EsriAPIKey`	<Enter ESRI API Key>		**Note** You can find your ESRI API Key on the dashboard in your ArcGIS developer account. For more information and API Key description, see ArcGIS Developers documentation.
CONFIG	`UseDynamicLayerDisplayPrefs`	TRUE/FALSE		Use layer visibility setting from each dynamic layer (business layer) as default.
CONFIG	`ZoomOnLocationEnabled`	TRUE/FALSE		Set to true if the map must zoom to current location when location is toggled on.
CONFIG	`EnableNearMe`	TRUE/FALSE		Enable near me feature
CONFIG	`EnableFeatureLayers`	TRUE/FALSE		Enable display of feature layers.
CONFIG	`EnableBaseMaps`	TRUE/FALSE		Enable display and switching of basemaps.
CONFIG	`EnableDynamicLayers`	TRUE/FALSE		Enable display of dynamic layers (business layers)

Parameter Group	Parameter Name	Parameter Value	Parent Parameter Group	Description
CONFIG	`EnableCurrentLocation`	TRUE/FALSE		Enable display of current location.
CONFIG	`DefaultNearMeRadius`	<number>		Default value of near me radius if Near Me feature is enabled in EnableNearMe parameter.
CONFIG	`DefaultNearMeUnits`	Mi/KM		Default value of near me radius unit if Near Me feature is enabled in EnableNearMe parameter.
CONFIG	`RoutingURL`	The following is an example of a URL put into your server ID: `https://utility.arcgis.com/usrsvcs/servers/<ID of server>/rest/services/World/Route/NAServer/Route_World`		Routing URL to be set if Field Operations Worker component (FOW) is enabled.
CONFIG	`GefUIProfileId`	<profile ID value as set in GEF config>		GEF profile id
CONFIG	`EnableGef`	TRUE/FALSE		Enable GEF as map data provider.

3.4.1.2 Configuring Your GIS Mapping Table

Use the /SMFND/GIS_OIDMAP_UPLOAD_PROG program to upload object values to the /SMFND/OID01_GS mapping table.

- **SPATIAL_OBJECTID:** Esri field

  > ⓘ Note
  >
  > You can maintain either the SPATIAL_OBJECTID field or the SPATIAL_GUID field, or both. However, it is recommended that you maintain the SPATIAL_OBJECTID field.

- **SPATIAL_GUID:** Esri field
- **OBJKEY:** SAP key (for example, FLOC or EQ, depending on OBTYP)
- **OBJECT_GROUP / OBJECT_GROUP1:** SAP fields that allow for extra attributes or segregation

3.4.1.3 Supporting Authenticated GIS Services

Use

You can view token-based authenticated basemaps and feature layers on the mobile client. Use the ConfigPanel to configure the client ID and client secret strings.

Configuring Authenticated GIS Services

The mobile client retrieves the tokens. The client ID and client secret are supplied to the client so each client can generate their tokens for accessing authenticated services.

If your organization wishes to access Esri application-level authenticated GIS services, configure the SAP Service and Asset Manager application as shown in the following procedure.

You can also configure a proxy through Esri. Authenticated basemaps and feature layers are requested through a local proxy. The proxy manages the generation and use of tokens based on the client ID and client secret. For more information on configuring a proxy, see the Esri documentation, Working with Proxy Services .

To turn on GIS authenticated services in the ConfigPanel, add the following rule:

1. From the ConfigPanel *Home* page, navigate to *OData Mobile Data Configuration* > *OData Mobile Data Object List* > *Data Filter Tab* > *SAP_ASSET_MANAGER_<XX>* > *SAM2310_GIS_MAP_CONTROL* > *Operation - READ* > *Data Distribution* > *INI_PARAMETER*.
2. Click the *Change* button. In the *Rule List* section, click the *Add* button to add a new rule. The rule gives you the freedom to retrieve your client credentials in a manner appropriate for your organization:
   - **Parameter Group:** `AUTHENTICATION`
   - **Parameter Name:** `ConfigRule`
   - **Parameter Value:** `/SAPAssetManager/Rules/Path/To/A/Rule.js`
     The specified rule must return a json object in the form of the following example. Be sure to replace the text in the sample code with your actual client ID and client secret (`"YourClientId"` and `"YourClientSecret"`):

     Sample Code

     ```
     {
         "ClientId": "YourClientId",
         "ClientSecret": "YourClientSecret"
     }
     ```

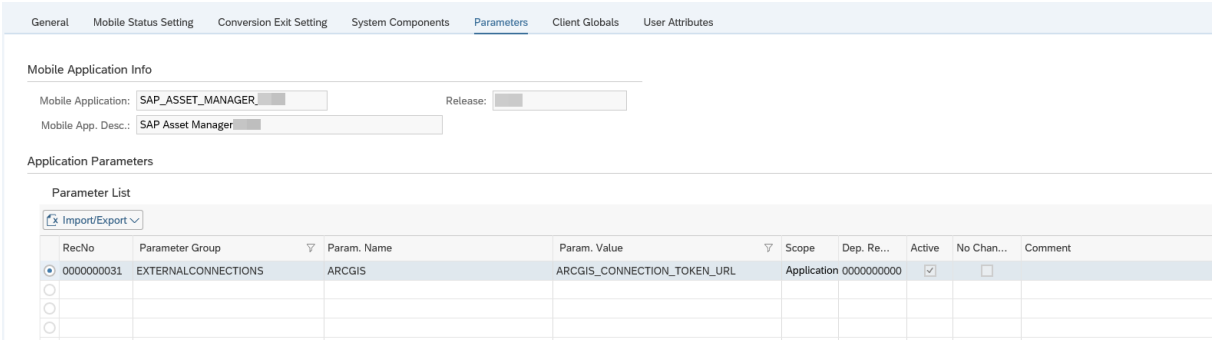
3.4.1.4 Creating and Configuring an RFC Destination for Offline Maps

Establish an oAuth connection using a secure TokenID to request offline maps information from the ArcGIS server.

Context

The oAuth 2.0 client ID and secret for your connection are stored in the RFC destination. Maintain the RFC destination in the app parameters in the ConfigPanel.

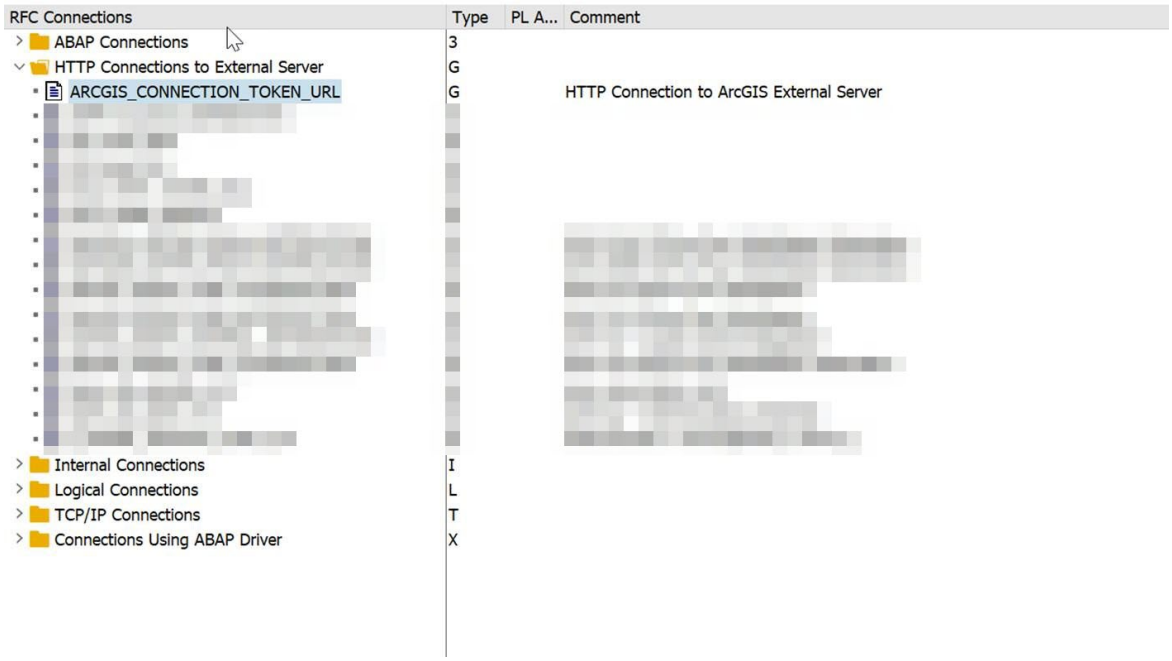
See the following screenshot for the parameter that is included in the out of the box SAP Service and Asset Manager application:



The *OauthTokens* entity is provided out of the box to use for the RFC destination set in the *Parameters* tab. Use the following procedure to create the *ARCGIS_CONNECTION_TOKEN_URL* RFC destination in your backend system:

Procedure

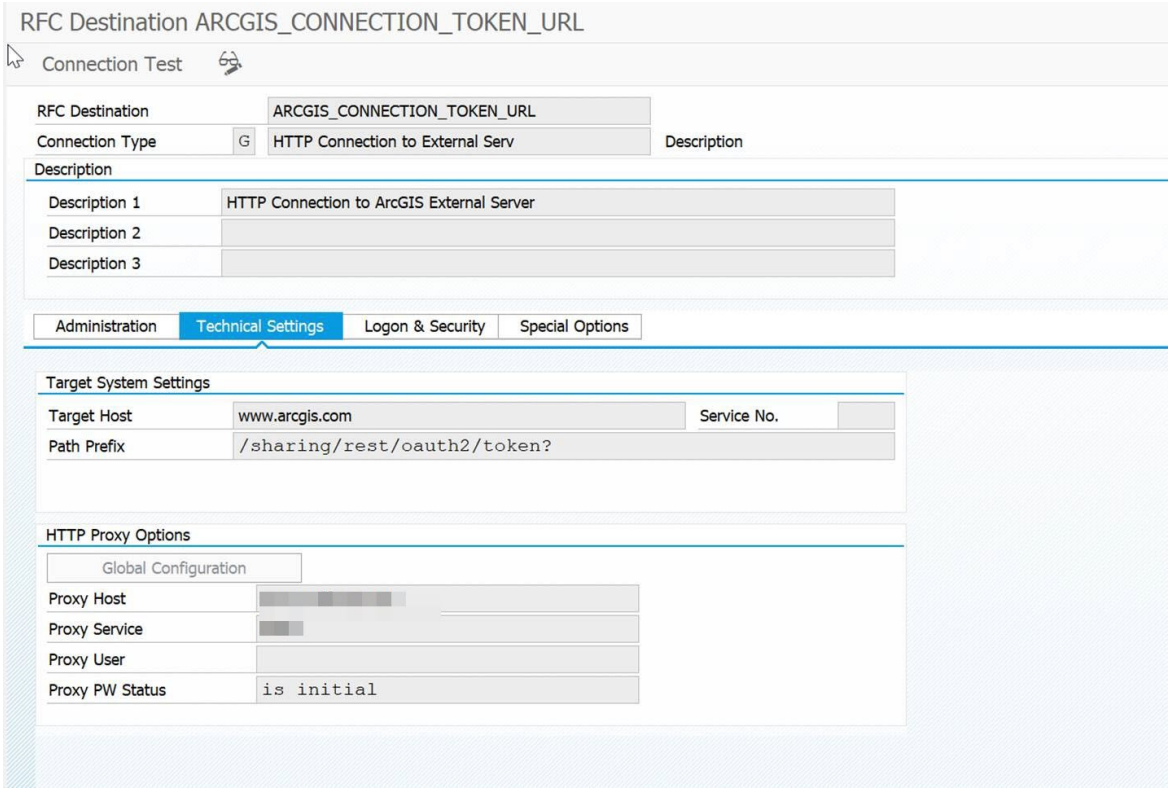
1. Using the SAP GUI enter transaction *SM59* and enter *ARCGIS_CONNECTION_TOKEN_URL*, the new RFC destination.



2. From the *Technical Settings* tab of your new RFC connection, set the *Target Host* to match the ArcGIS server URL. Find the information for the `Target Host` field in the `Service Host` field on the *General Data* tab of the *Geospatial Service Definitions* page in the ConfigPortal. Set the *Path Prefix* to `/sharing/rest/oauth2/token?`.

   > ⓘ Note

3. If necessary, configure the proxy that you're using to allow your back end systems to connect to the internet.



4. Click the *Logon & Security* tab. Under *Logon Procedure*, select *Basic Authentication*. In the *User Name* field, enter the client ID. In the *Password* field, enter the client secret.
5. In the *Security Options* section of the *Logon & Security* tab, ensure that the SSL is set to *Active*.
6. Select your SSL certificate from the *SSL Certificates* list.
7. *Save* the connection and perform a connection test.

   If the connection is configured correctly, a 200 HTTP response is returned.
8. Import the SSL certificate and ensure its active using transaction code *STRUST*.

3.4.1.4.1 Configuring GIS Offline Metadata

Create a new rule in the SAP Service and Asset Manager metadata to retrieve a valid authentication token for authenticated resources within the map control.

Context

To create a new rule for offline GIS, see the existing rule found in the metadata file of SAP Service and Asset Manager at `/SAPAssetManager/Rules/Extensions/TokenRequest.js`. The rule found there uses an online oData request to get the token from the back-end server. This rule is the recommended way to retrieve a token from the Esri oAuth service.

Procedure

1. Navigate to the metadata of your SAP Service and Asset Manager project using your preferred text editor or the SAP Web IDE and create a new rule.
2. Save the rule you created.
3. Open the `.page` file that contains your map control using your preferred text editor.
4. Navigate to the *Controls* section and locate the control for the map. The map control has the following specific properties:
   - *Type* of *Control.Type.Extension*
   - *Module* of *Extension-MapFramework*
5. Add a *TokenAuthentication* object within the *ExtensionProperties* object of the control. Add an *Action* property of a rule to retrieve a valid authentication token as shown in the following example:

   Sample Code

   ```
   {
            "_Type":"Control.Type.Extension",
            "Module": "extension-MapFramework",
            "Control":"MapExtensionWithContext",
            "Class":"MapExtensionWithContext",
            "_Name":"MapExtensionControl",
            "ExtensionProperties": {
                 …
                   "TokenAuthentication": {
                                  "Action": "/path/to/rule/that/returns/a/
   token"
                   },
                 …
            }
       }
   ```

6. Save the page and update your application with the new metadata.

Results

The new metadata allows token authentication on your ArcGIS map when required.

3.4.2 Geographical Enablement Framework (GEF)

Geographical Enablement Framework (GEF) works as the foundation to extend business data with geometric attributes for SAP S/4HANA. GEF is integrated directly into your SAP S/4HANA system.

As a framework leveraging the spatial capabilities inherent on the SAP S/4HANA platform, it enables organizations to develop geospatially enriched business data, and make them accessible from within SAP Service and Asset Manager.

SAP Service and Asset Manager uses the geometries (points, lines, and polygons) from the GEF geotables stored in SAP S/4HANA for the geo-enabled objects in the application. Users can view and work with base maps and feature layers on the device.

> **Note**
>
> Esri base maps are the only base maps supported for the app.

For detailed information about GEF, including installation and implementation procedures, see the Geographical Enablement Framework documentation.

For information about configuring GIS, see the subtopics located in the Geographic Information System (GIS) [page 197] topic.

3.4.2.1 Disabling GIS

SAP Service and Asset Manager utilizes GEF by default. When implementing GEF, ensure that GIS is disabled in the ConfigPanel.

1. From the ConfigPanel home page, navigate to *Geospatial Service Definitions*. Select your application release using the *Mobile Application Filter*. Expand the *Geospatial Mobile Services by App* tree in the left panel.
2. Uncheck the *Active* boxes from the following services:
   - *<APPXX>_GIS_QUERY_EQUIPMENT*
   - *<APPXX>_GIS_QUERY_FLOC*
   - *<APPXX>_GIS_QUERY_NOTIFICATION*
   - *<APPXX>_GIS_QUERY_WORKORDER*

3.4.2.2 Configuring Your Map Settings - GEF

Some GEF settings are standard with the initial SAP Service and Asset Manager application.

You can change any of the settings described in this topic to configure the application for your site.

> **Note**
>
> You can also change the map setting metadata through the Mobile Development Kit. Note that if there are metadata differences, Mobile Development Kit changes override ConfigPanel changes.

Before enabling GEF, deactivate GIS if you previously had it activated. See the Disabling GIS [page 207] topic for more information.

Geospatial Service Definitions - General Data Tab

1. From the ConfigPanel home page, navigate to *Geospatial Service Definitions*. Select your application release using the *Mobile Application Filter*. Expand the *Geospatial Mobile Services by App* tree in the left panel.
2. Check the *Active* boxes from the following services:
   - *<APPXX>_GEF_QUERY_EQUIPMENT*
   - *<APPXX>_GEF_QUERY_FLOC*
   - *<APPXX>_GEF_QUERY_NOTIFICATION*
   - *<APPXX>_GEF_QUERY_WORKORDER*

   > **Note**
   >
   > If you are using custom GEF services, enable those rather than the standard GEF services.
3. *Save* your changes.

Geospatial Service Definitions - Parameters Tab

If you've mapped a custom business object to a GEF scenario using *SPRO* in the SAP GUI, you must also map the connection in the Config Panel.

1. From the Config Panel home screen, navigate to *Geospatial Service Definitions* > *Parameter Settings tab*. Select your application release, then select the GEF service you've customized in the *Geospatial Services by Mobile App* list.
2. Click the *Change* button.
3. In the *Operation Parameter Settings* section, navigate to *Parameters for Service Operation* > *Provider Operation - QUERY* > *Standard Parameter* > *Parameter - EAM Scenario*.
4. In the *Value Setting* section, change the *Parameter Value* to your custom business object.
5. *Save* your changes.

OData Mobile Data Object Configuration - Data Filter Tab

1. From the ConfigPanel Home page, navigate to *OData Mobile Data Object Configuration* > *Data Filter Tab* > *<SAMXX>_GIS_OBJECT_DATA* > *Operation - READ* > *Standard Filter* > *GEOSERVICE_ID*. Click the *Change* button.
   Ensure that any GIS objects are inactive. Ensure that the following objects are active:
   - *<APPXX>_GEF_QUERY_EQUIPMENT*
   - *<APPXX>_GEF_QUERY_FLOC*
   - *<APPXX>_GEF_QUERY_NOTIFICATION*
   - *<APPXX>_GEF_QUERY_WORKORDER*

> ### Note
>
> If you have any custom OMDOs for GEF, enable those. If you have any custom OMDOs for GEF, enable those, rather than the standard GEF OMDOs.

2. While still on the *Data Filter* tab with the *<SAMXX>_GIS_OBJECT_DATA* selected, navigate to *Operation - READ* > *Data Distribution* > *GIS_ASSIGNMENT_TYPE*.
   Ensure the *Range Value* is set to *2 - GEF Integration*.
3. By default, all basemaps are maintained. GEF uses information coming from SAP S/4HANA GEF APIs. When you activate GEF, all GIS configuration is invalid. Configure the following parameters for GEF:
   While still on the *Data Filter* tab, navigate to *<SAMXX>_GIS_MAP_CONTROL* > *Operation - READ* > *Data Distribution* > *INI_PARAMETER*.
   Ensure that the following parameters found in the *CONFIG* parameter group are configured for GEF:
   - **GefUIProfileID**: Standard configuration is *EAMALL*. If you've customized this in the SAP GUI, select your customization.
   - **EnableGef**: Set to *True*

Save your changes.

3.4.2.3 Configuring Pagination

You can configure the number of business objects with geometry information displayed per page.

Procedure

1. Navigate to `metadata/definitions/Pages/Extensions/Map.page`.
2. Change `100` to the number of business objects displayed per page as shown in the following example:

> ### Sample Code
>
> ```
>    "Controls": [
>                "ExtensionProperties": {
>                "ItemsPerPage": 100,
> ```

3.4.2.4 Enabling GEF Create and Update

SAP Service and Asset Manager supports GEF geometry download as well as GEF create and update.

Context

- **Create:** Supports the ability to create GEF point, line, and poly-line directly from the map and from the MAIF `/MERP/CL_CORE_GIS_GEOSERV_GEF` object handler.
- **Update:** Supports the ability to update GEF point, line, and poly-line directly from the map and from the MAIF `/MERP/CL_CORE_GIS_GEOSERV_GEF` object handler.

> **Note**
>
> The update feature supports the ability to update a time-sensitive GEF coordinate. Currently, the create feature does not support time-sensitive geometry.

Procedure

1. In the ConfigPanel, navigate to the *OData Mobile Data Object Configuration* section and find OMDO **SAM2310_GIS_OBJECT_DATA**. Click the *Data Filter* tab.
2. Click the *Change* button.
3. Expand the *Defined Filters* list to show and change the following ▶ ▶ ▼.
   a. ▶ *Operation - CREATE* ▶ *Standard Filter* ▶ *GEOSERVICE_ID** ▼. Perform the following actions:
      - Uncheck the *Active* flag for all GIS rules.
      - Check the *Active* flag for all GEF rules.
   b. ▶ *Operation - READ* ▶ *Standard Filter* ▶ *GEOSERVICE_ID** ▼. Perform the following actions:
      - Uncheck the *Active* flag for all GIS rules.
      - Check the *Active* flag for all GEF rules.
   c. ▶ *Operation - UPDATE* ▶ *Standard Filter* ▶ *GEOSERVICE_ID** ▼. Perform the following actions:
      - Uncheck the *Active* flag for all GIS rules.
      - Check the *Active* flag for all GEF rules.
4. Staying in the **SAM2310_GIS_OBJECT_DATA** oMDO, navigate to ▶ *Operation - READ* ▶ *Data Distribution* ▶ *GIS_ASSIGNMENT_TYPE** ▼. Select *2 - GEF Integration* and be sure the *Active* flag is checked.
5. Move to the **SAM2310_GIS_MAP_CONTROL** oMDO and navigate to ▶ *Operation - READ* ▶ *Data Distribution* ▶ *INI_PARAMTER** ▼. Check the *Active* flag for the *EnableGef* parameter found in the *CONFIG* parameter group.
6. From the ConfigPanel home page, navigate to *Geospatial Service Definitions*. Select your application release using the *Mobile Application Filter*. Expand the *Geospatial Mobile Services by App* tree in the left panel. Ensure that the following services are active in the *General* tab:

- SAM2310_GEF_ADD_EQUIPMENT
- SAM2310_GEF_ADD_FLOC
- SAM2310_GEF_ADD_NOTIFICATION
- SAM2310_GEF_ADD_WORKORDER
- SAM2310_GEF_UPDATE_EQUIPMENT
- SAM2310_GEF_UPDATE_FLOC
- SAM2310_GEF_UPDATE_NOTIFICATION
- SAM2310_GEF_UPDATE_WORKORDER

7. *Save* your changes.

3.5 Push Framework Settings Procedures

3.5.1 Configuring Push for Work Order Assignment Type 1

Context

> **Note**
>
> Push configuration is available for SAP Service and Asset Manager for Android starting with the 4.0 release.

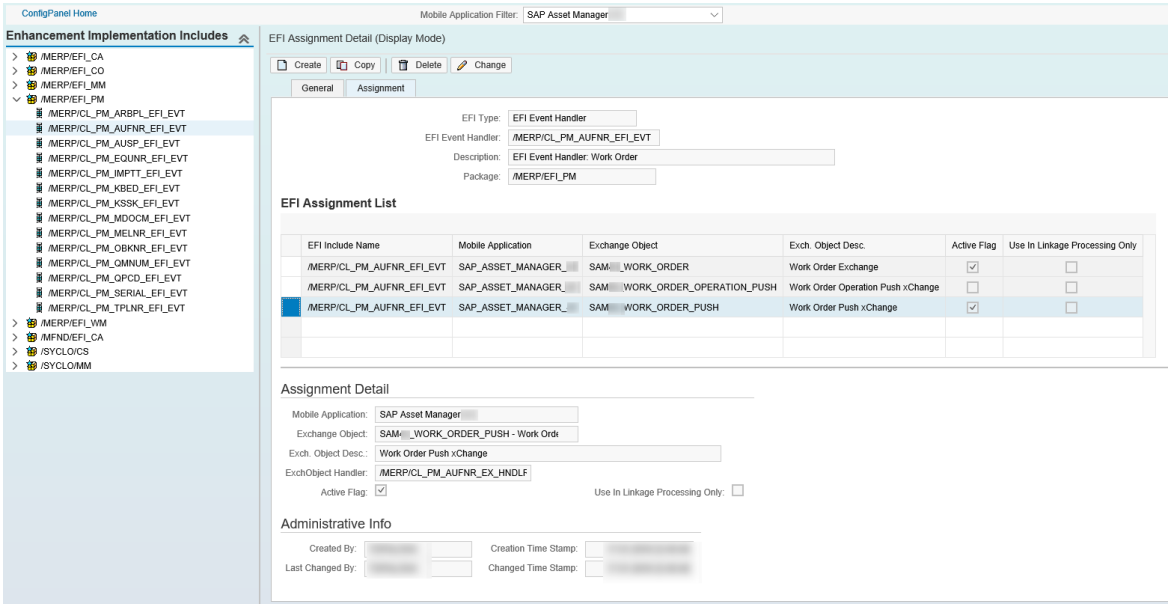
Event-based push is supported for assignment types 1 (header-level person responsible for the work order) and 2 (operation-level personnel number of the work order). You can only configure push for one work order assignment type at a time.

By default, work order push is enabled for whichever assignment type your work order OMDO is set to. You can manually assign the *WO_ASSIGNMENT_TYPE* filter for your data distribution model to *1* or *2* to set a push assignment type different to the assignment type of the work order data distribution.

Procedure

1. In the ConfigPanel, navigate to the *OData Mobile Data Object Configuration* section and find OMDO **SAM2310_WORK_ORDER_GENERIC**. Ensure the filter *WO_ASSIGNMENT_TYPE* is set to *1* on the *Data Filter* tab.
2. Return to the *Home* page of the ConfigPanel. Click the *Push Scenario Definition* link. Ensure that your mobileapplication is selected in the *Mobile Application Filter*.
3. Ensure the *Active* flag for the **SAM2310_EMERGENCY_WORKORDER_PUSH** scenario on the *General Data* tab is checked. Note that you can have both work order and notification pushes marked as Active because they are separate objects.
4. Make sure the configuration in the *Source Setting* and *Distribution Setting* sections are correct.
   By default, the `<Source Object>` for the work order operation push is the exchange object **SAM2310_WORK_ORDER_PUSH** with the `<Distribution Object>` **SAM2310_WORKORDER_PUSH**.

5. Return to the ConfigPanel Home page, then navigate to the *EFI Assignment* section. In the *Enhancement Implementation Includes* list, select */MERP/EFI_PM* > */MERP/CL_PM_AUFNR_EFI_EVT*.
6. Click the *Assignment* tab. Ensure the *Active* checkbox is checked for the exchange object **SAM2310_WORK_ORDER_PUSH**.



7. *Save* your changes.

3.5.2 Configuring Push for Work Order Assignment Type 2

Context

> **Note**
>
> Push configuration is available for SAP Service and Asset Manager for Android starting with the 4.0 release.

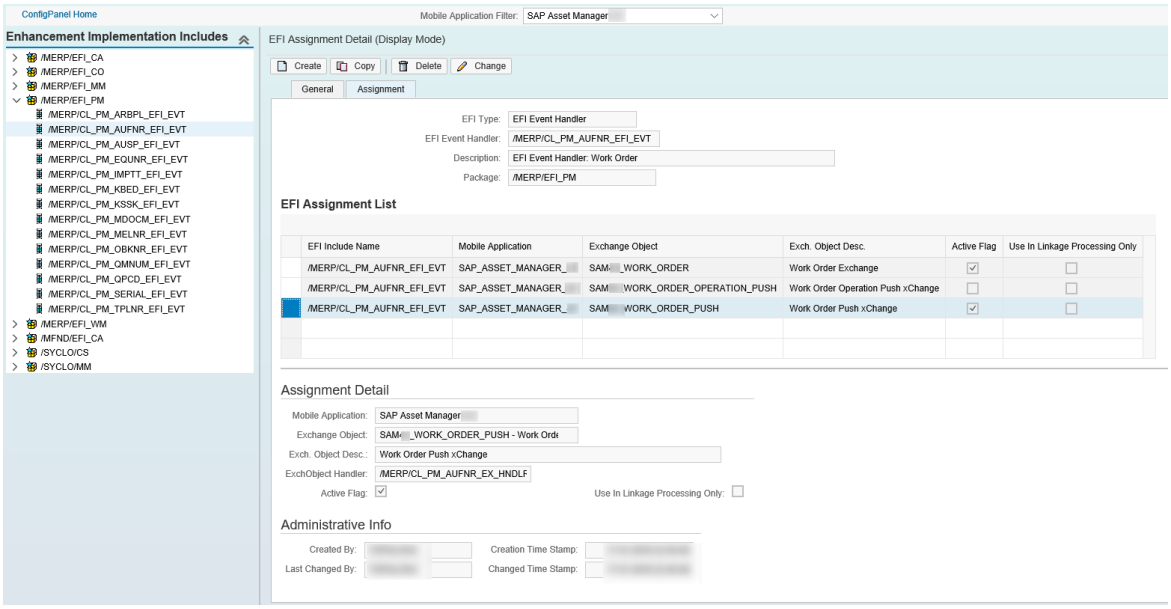
Event-based push is supported for assignment types 1 (header-level person responsible for the work order) and 2 (operation-level personnel number of the work order). You can only configure push for one work order assignment type at a time.

By default, work order push is enabled for whichever assignment type your work order OMDO is set to. You can manually assign the *WO_ASSIGNMENT_TYPE* filter for your data distribution model to *1* or *2* to set a push assignment type different to the assignment type of the work order data distribution.

Procedure

1. In the ConfigPanel, navigate to the *OData Mobile Data Object Configuration* section and find OMDO **SAM2310_WORK_ORDER_GENERIC**. Ensure the filter *WO_ASSIGNMENT_TYPE* is set to *2* on the *Data Filter* tab.

2. Return to the *Home* page of the ConfigPanel. Click the *Push Scenario Definition* link. Ensure that your mobileapplication is selected in the *Mobile Application Filter*.
3. Ensure the *Active* flag for the **SAM2310_EMERGENCY_WORKORDER_PUSH** scenario on the *General Data* tab is checked. Note that you can have both work order and notification pushes marked as Active because they are separate objects.
4. Make sure the configuration in the *Source Setting* and *Distribution Setting* sections are correct.
   By default, the `<Source Object>` for the work order operation push is the exchange object **SAM2310_WORK_ORDER_PUSH** with the `<Distribution Object>` **SAM2310_WORKORDER_PUSH**.
5. Return to the ConfigPanel Home page, then navigate to the *EFI Assignment* section. In the *Enhancement Implementation Includes* list, select *> /MERP/EFI_PM > /MERP/CL_PM_AUFNR_EFI_EVT >*.
6. Click the *Assignment* tab. Ensure the *Active* checkbox is checked for the exchange object **SAM2310_WORK_ORDER_OPERATION_PUSH**.



7. *Save* your changes.

3.5.3 Configuring Push for Notification Assignment Types 1 - 5

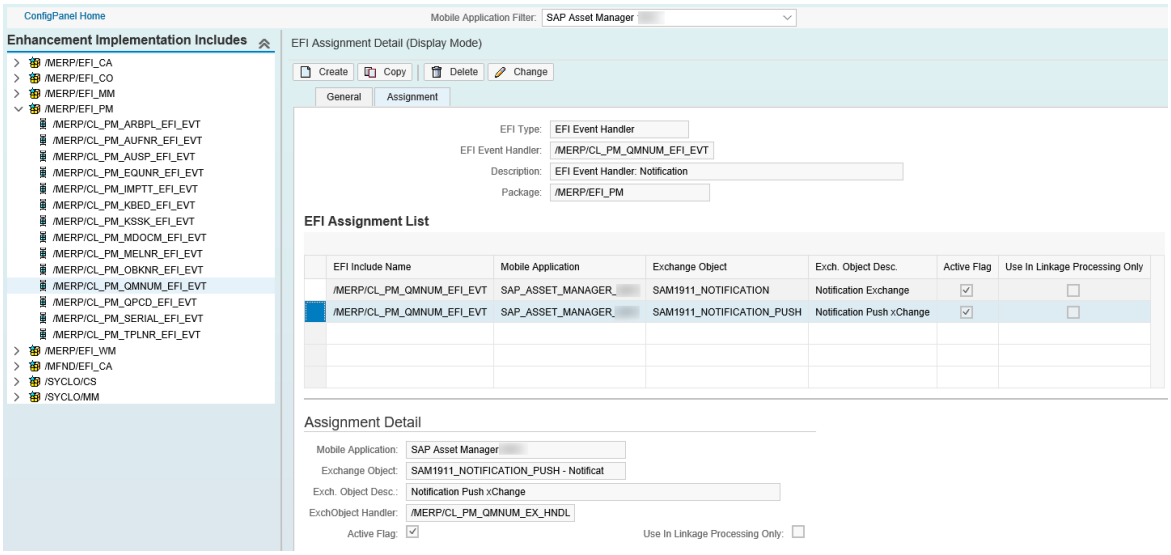
Context

Event-based push is supported for notification assignment types 1 through 5. You can only configure push for one notification assignment type at a time.

By default, notification push is enabled for whichever assignment type your notification OMDO is set to. You can manually assign the *NOTIF_ASSIGNMENT_TYPE* filter for your data distribution model to *1*, *2*, *3*, *4* or *5* to set a push assignment type different to the assignment type of the notification data distribution.

Procedure

1. In the ConfigPanel, navigate to the *OData Mobile Data Object Configuration* section and find OMDO **SAM2310_NOTIFICATION_GENERIC**. Ensure the filter *NOTIF_ASSIGNMENT_TYPE* is set to the assignment type of your choice (*1* - *5*) on the *Data Filter* tab.
2. Return to the *Home* page of the ConfigPanel. Click the *Push Scenario Definition* link. Ensure that your mobile application is selected in the *Mobile Application Filter*.
3. Ensure the *Active* flag for the **SAM2310_EMERGENCY_NOTIFICATION_PUSH** scenario on the *General Data* tab is checked. Note that you can have both work order and notification pushes marked as Active because they are separate objects.
4. Make sure the configuration in the *Source Setting* and *Distribution Setting* sections are correct. By default, the `<Source Object>` for the notification push is the exchange object **SAM2310_NOTIFICATION_PUSH** with the `<Distribution Object>` **SAM2310_NOTIFICATION_PUSH**.
5. Return to the ConfigPanel Home page, then navigate to the *EFI Assignment* section. In the *Enhancement Implementation Includes* list, select *⏵ /MERP/EFI_PM ⏵ /MERP/CL_PM_QMNUM_EFI_EVT ⏵*.
6. Click the *Assignment* tab. Ensure the *Active* checkbox is checked for the exchange object **SAM2310_NOTIFICATION_PUSH**.



7. *Save* your changes.

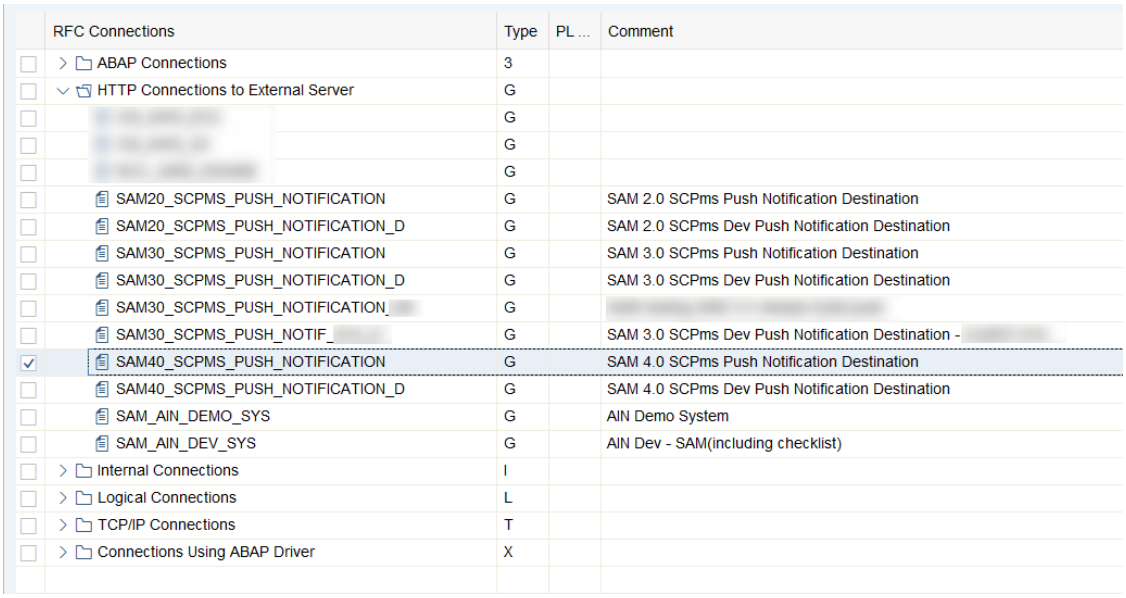
3.5.4 Setting up the Outbound Trigger for your Push Configuration

Context

> **Note**
>
> Push configuration is available for SAP Service and Asset Manager for Android starting with the 4.0 release.

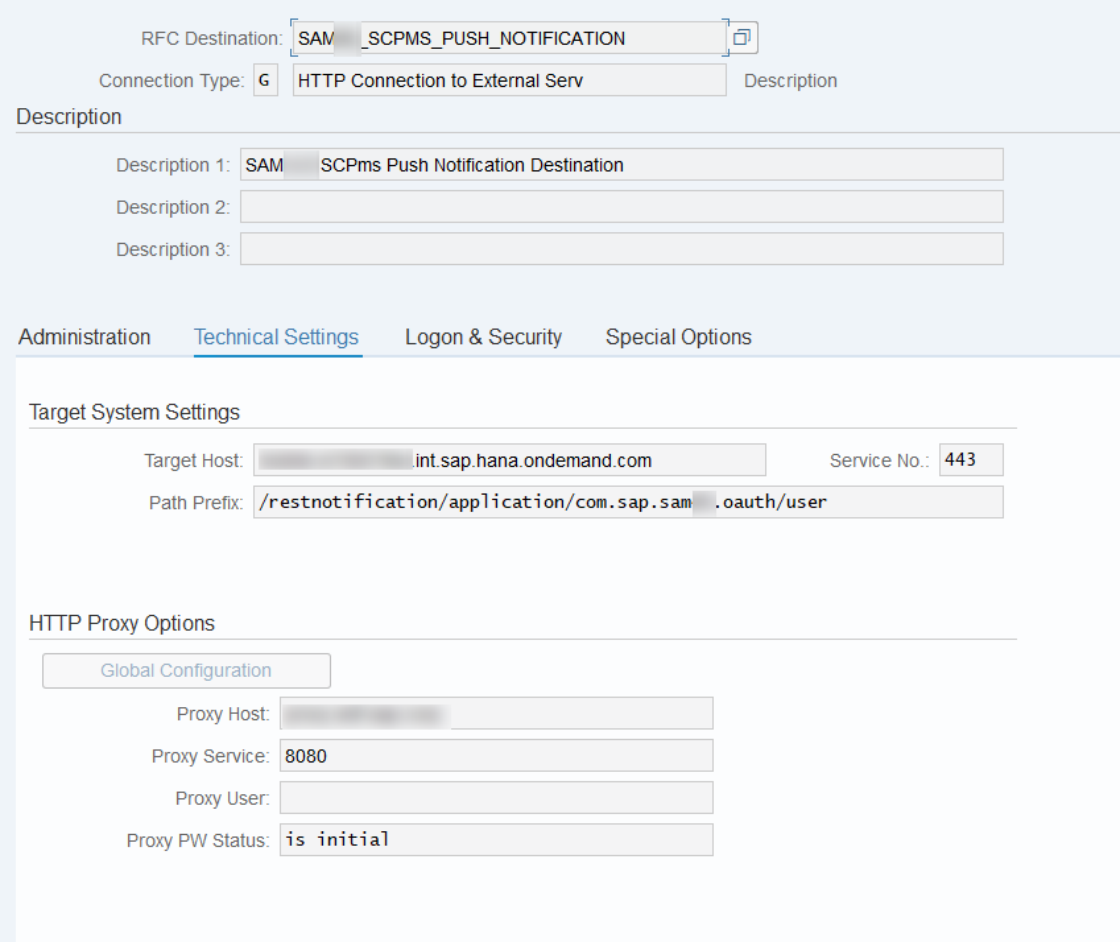
Procedure

1. From the ConfigPanel *Home* page, navigate to the *Outbound Trigger Configuration* section and select your desired mobile application from the *Mobile Application Filter* dropdown menu at the top of the page.
2. From the *Outbound Triggers by Mobile App* list, select the outbound trigger **SAM2310_WORKORDER_TRIGGER_SCPMS**. Make sure that the `<Cloud Platform Mobile App ID>` matches your mobile services application ID from SAP Business Technology Platform Mobile Services. By default, the application ID is set to *com.sap.<appXX>.oauth.prod*.
3. Set up the RFC destination **SAM2310_SCPMS_PUSH_NOTIFICATION** pointing to the mobile services host name using the SAP GUI:
   a. In the SAP GUI, using transaction *SM59*, add the following new RFC destination: **SAM2310_SCPMS_PUSH_NOTIFICATION** of type *G (HTTP Connection to External Serv)*



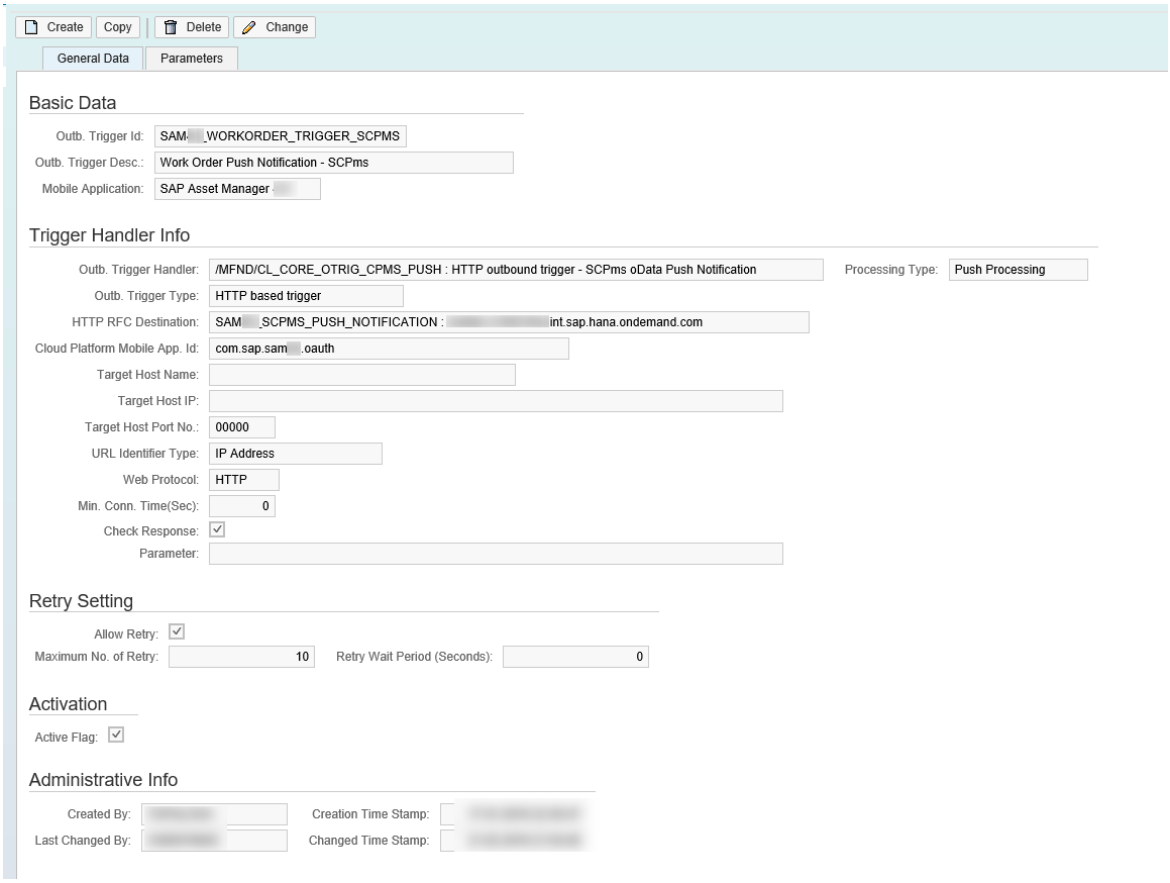
   b. On the *Technical Settings* tab of the new connection, set the *Target Host* to match the push API of the SAP Business Technology Platform Mobile Services. Use service number *443*, which is the port number of the HTTPS connections.

> **Note**
>
> If necessary, configure the proxy that you are using to allow your back-end systems to connect to the Internet.



c. On the *Logon & Security* tab, under the *Logon Procedure*, select *Basic Authentication*. Enter the user name and password of the service user. Assign the *Notification User* role to this service user in the SAP Business Technology Platform Mobile Services configuration to ensure that the user is allowed to utilize the push notification service.
d. In the *Security Options* section of the *Logon & Security* tab, ensure that the *SSL* is set to *Active*.
e. In the *Security Options* section, ensure that the *SSL Certificate List* contains the SAP Business Technology Platform certificate chain. You can check the certificate chain at the SSL client certificate list with the *STRUST* transaction. Display the SAP Business Technology Platform Mobile Services certificate chain in any web browser by opening the SAP Business Technology Platform Mobile Services cockpit and checking the security settings.
f. Save the connection and perform a connection test. If the configuration is completed properly, a 200 HTTP response is returned.
g. In the *Technical Settings* tab, add the rest of the path for the PUSH API in *Path Prefix* to allow the POST operation to work properly.

4. In the ConfigPanel, return to the *Outbound Trigger Configuration* panel, *General Data* tab, if you are not already there. Check the outbound trigger configuration to ensure that the previously established *HTTP RFC Destination* is set as the RFC destination to the proper mobile application. This configuration ensures that push notifications reach the right mobile application user.



5. On the *Parameters* tab, set parameter *SCPMS_WITH_SAP_USER_ID* to *True* if the SAP Business Technology Platform user store is the same as the back end user store, that is, if you use the same mobile user for both SAP Business Technology Platform logon and back end logon. If the user store is not identical, set the parameter to *False* and update the mobile user setting as explained in the next step.
6. If parameter *SCPMS_WITH_SAP_USER_ID* is set to *False*, maintain the user mapping under *User Management* using the Admin portal in the SAP GUI:

> ### Note
>
> On the first delta sync, the SAP Service and Asset Manager client automatically performs substeps *a-d* for you. If desired, you can still perform these substeps to verify that the push registration process has completed successfully.

   a. Using the SAP GUI, launch the Admin portal with transaction code */n/SYCLO/ADMIN*. On the Admin portal home page, select *Administration* > *User Management*. Make sure to select your desired mobile application in the *Mobile Application Filter* field at the top of the page. Choose *Search* to list all users for that application.
   b. Select *User ID* under *Search Result*, and click the *Client Registration Info* tab under the *Mobile User Detail* section. Choose *Change* from the menu bar.
   c. Enter the matching *CPms User Id* (using upper case) for the back end user name listed under this tab.
   d. *Save* your changes.
7. Return to the *Home* page of the ConfigPanel. Select the *Push Scenario Definition* page. Under *Push Scenarios by Mobile App* list, select the desired push scenario definition. Click the *Outbound Trigger* tab and ensure that the proper outbound trigger is assigned and active for the push scenario.

8. *Save* your changes.

3.5.5 Activating Default Push Services for SAP Service and Asset Manager - Neo

Prerequisites

> **Note**
>
> The *SAP Service and Asset Manager Installation Guide* is a guide to setting up the basic framework necessary for push services using the default settings. For more details regarding configuration of push services, see the topic.

Before performing the procedure, ensure the following:

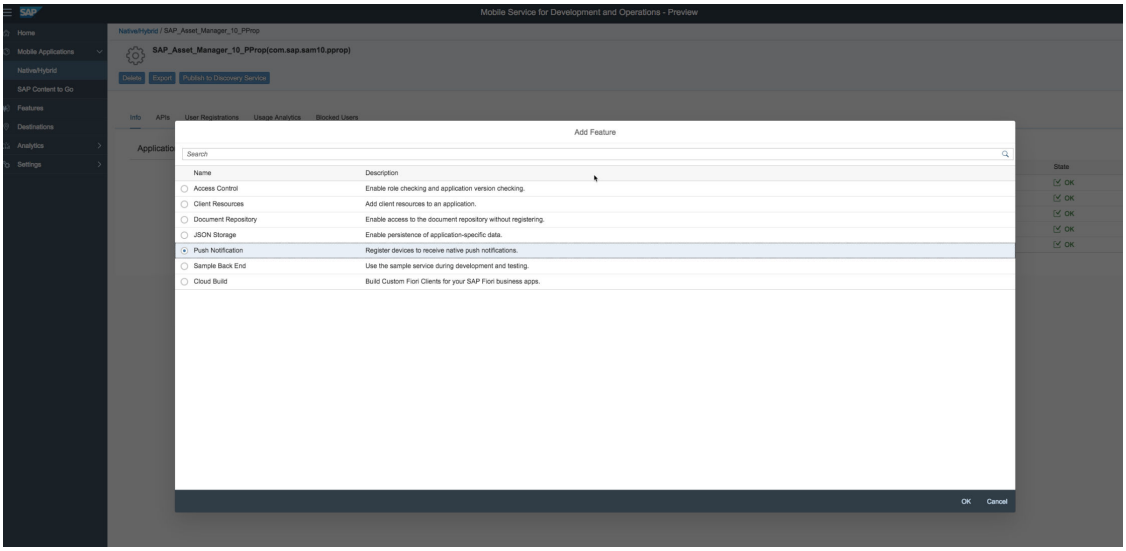
- SAP Service and Asset Manager application on the device is running on Mobile Development Kit 2.2.001
- You have installed SAP Service and Asset Manager 2310
- You have installed either SAP Mobile Add-On ECC or SAP Mobile Add-On for SAP S/4HANA. See the following installation guides on the following portal pages for version information:
  - SAP S/4HANA Mobile Add-On
  - SAP Mobile Add-On ECC

Context

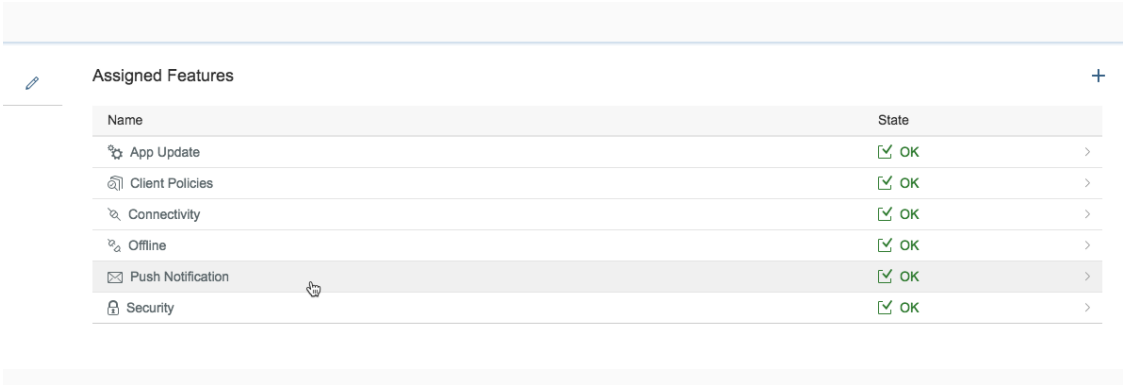
To configure and activate push for the Android platform, see the Configuring Push for Android [page 234] procedure.

Procedure

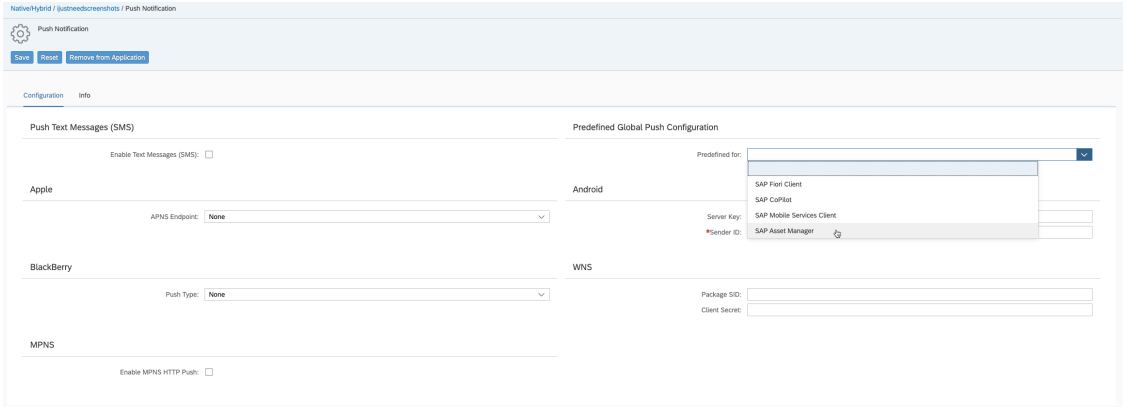
1. **Configure the SAP Business Technology Platform Mobile Services push API:**
   a. Enable the *Push Notification* feature in SAP BTP services:



b. Navigate to the *Push Notification* in the mobile app and create a new HTTP connection to your external server named *Assigned Features*:



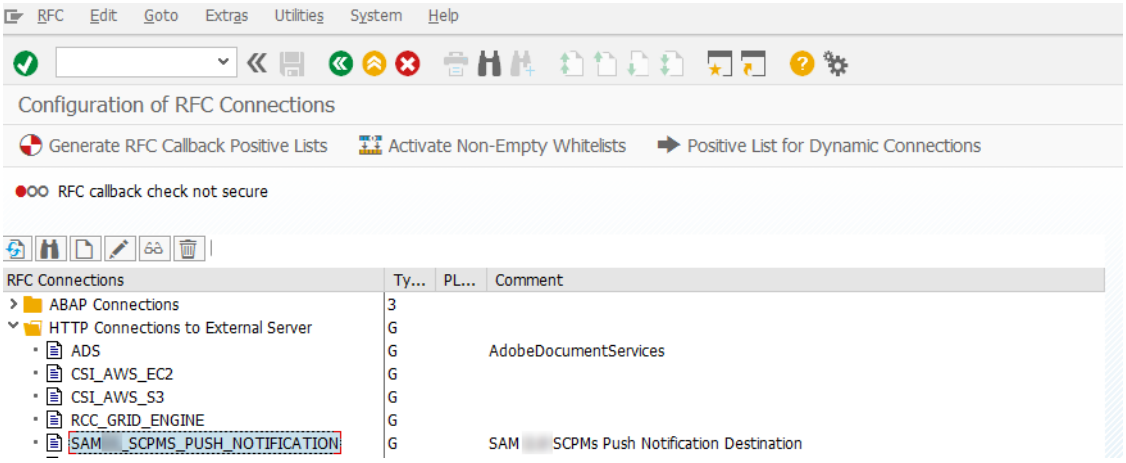
c. If you are using a custom deployment of SAP Service and Asset Manager, upload the corresponding APNs certificates here. If you are using the default application provided by the Apple App or the Google Play store, select *Predefined for* SAP Service and Asset Manager in the *Predefined Global Push Configuration* section.



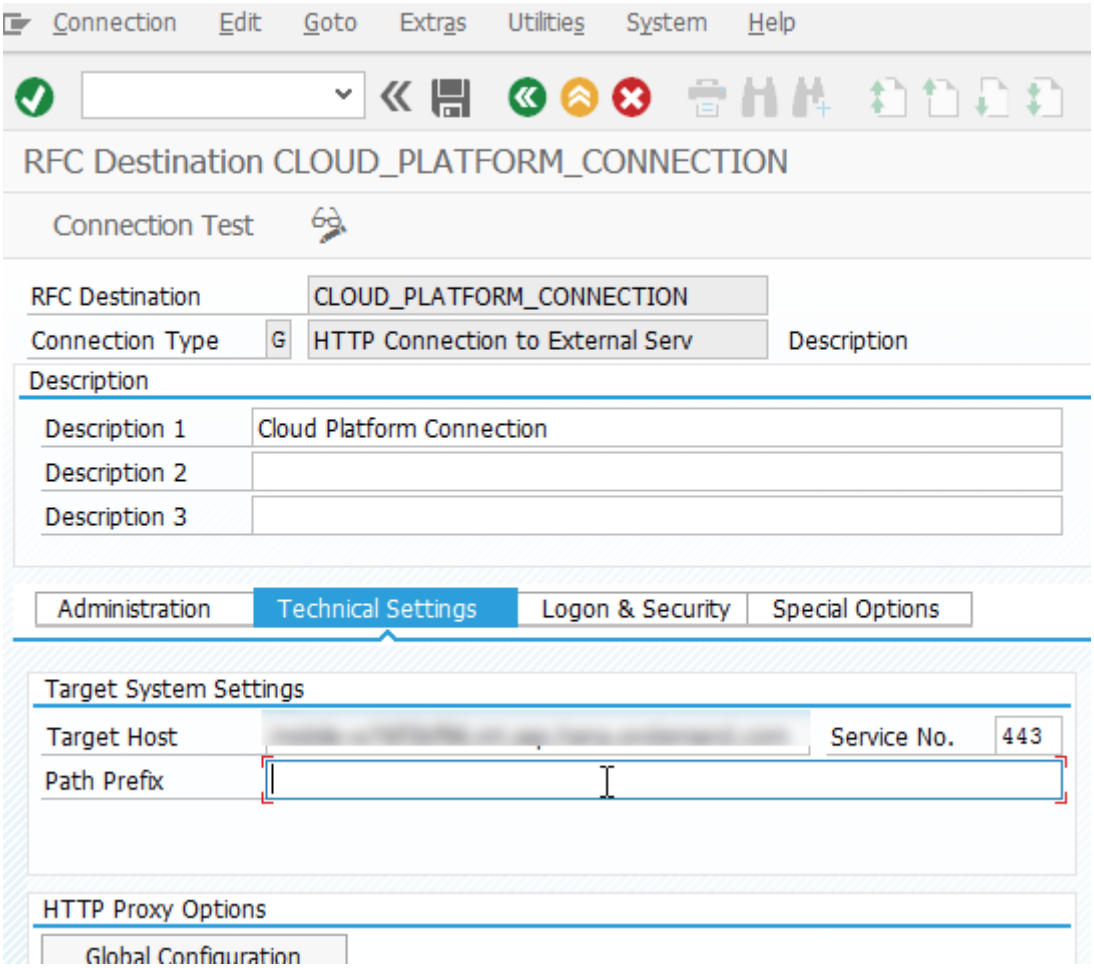
The SAP Business Technology Platform Mobile Services is configured for push.

2. **Configure the back-end system to utilize the SAP BTP services push APIs:**

a. Using the SAP GUI, run transaction *SM59*. Create a new HTTP connection with the name *SAM2310_SCPMS_PUSH_NOTIFICATION*.



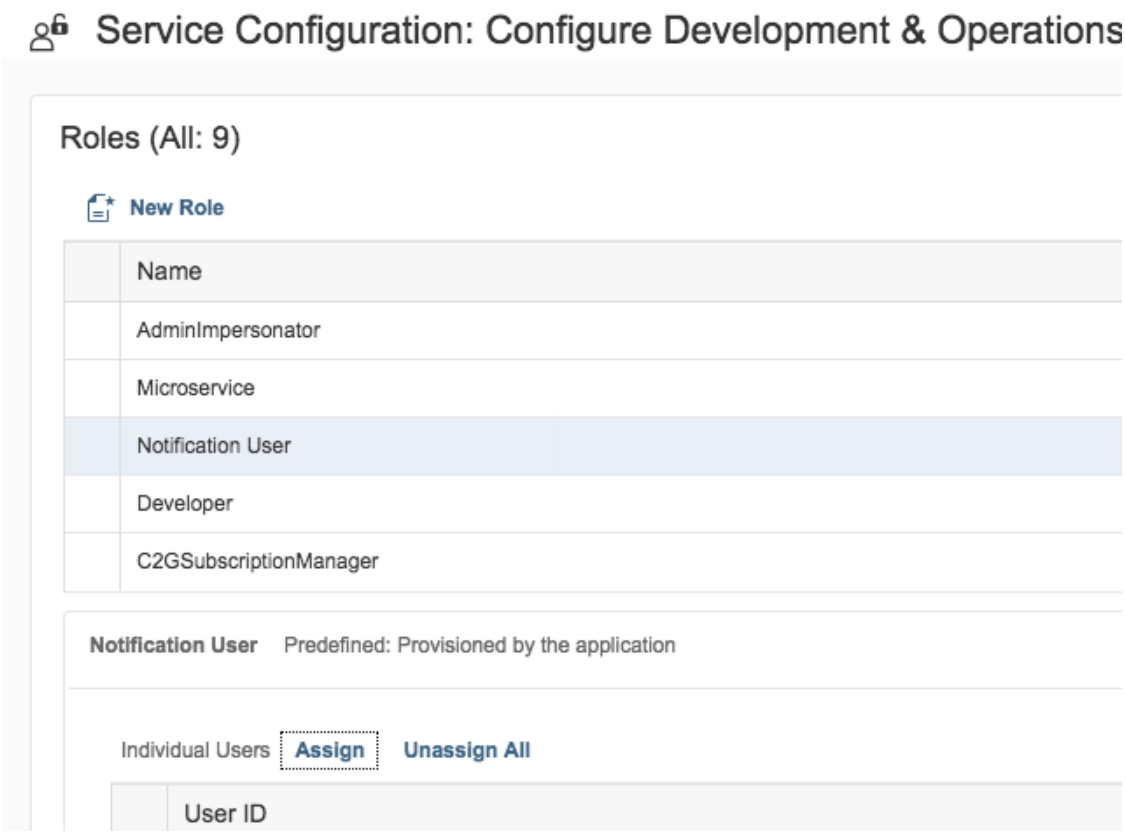
b. In the *Technical Settings* and tab of the new connection, set the *Target Host* to match the push API of the SAP BTP services, using *443* (the port number for HTTPS connections).



> **Note**
>
> If necessary, configure the proxy you are using to allow your back end to connect to the outside internet.

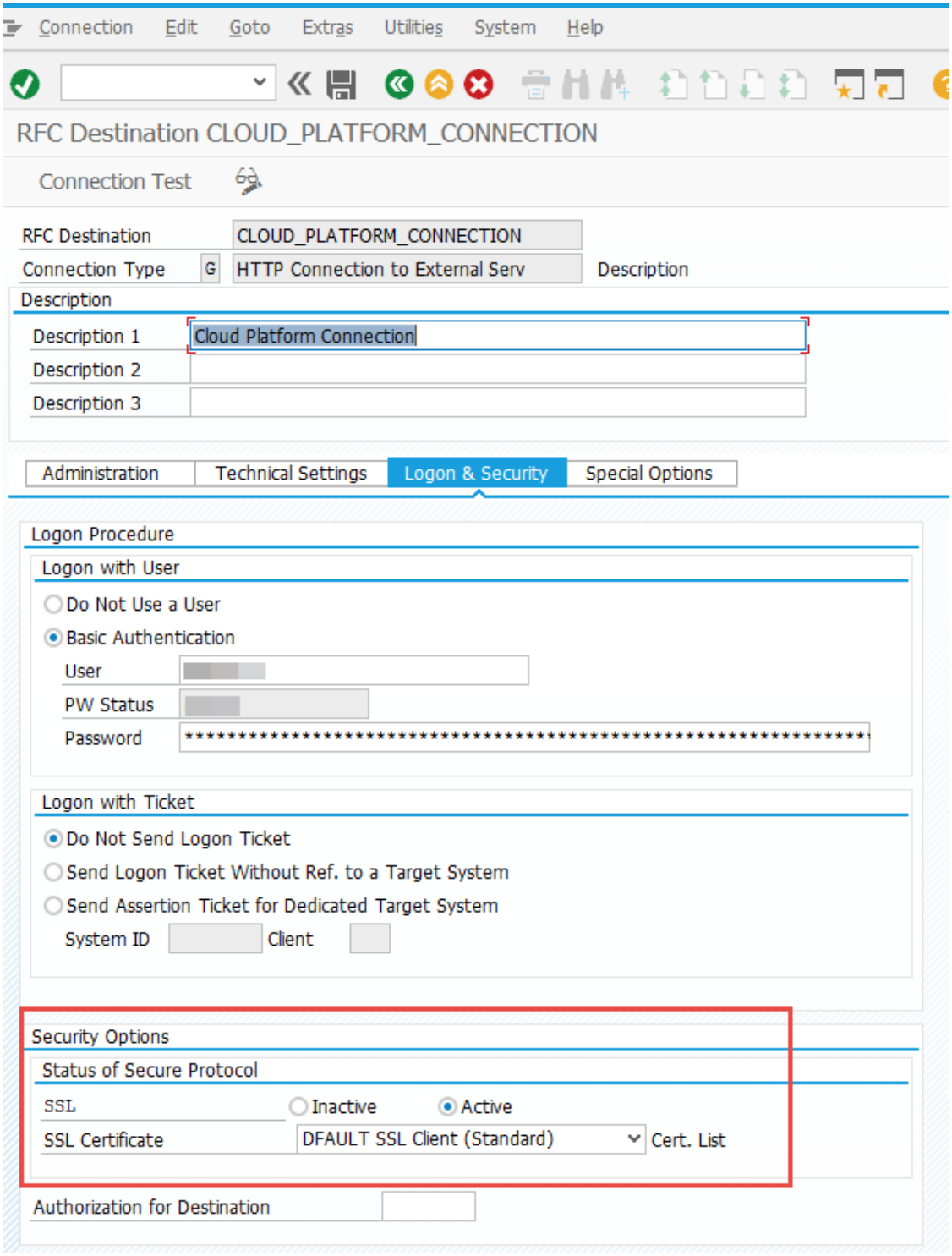
c. Click the *Logon & Security* tab. Under the *Logon Procedure*, select *Basic Authentication*. Enter the user name and password of a service user.

d. In SAP Business Technology Platform, ensure your service user has the role of *Notification User* assigned to them to ensure that the user is allowed to utilize the SAP BTP services API. The service user must be a member of the SAP Business Technology Platform account.

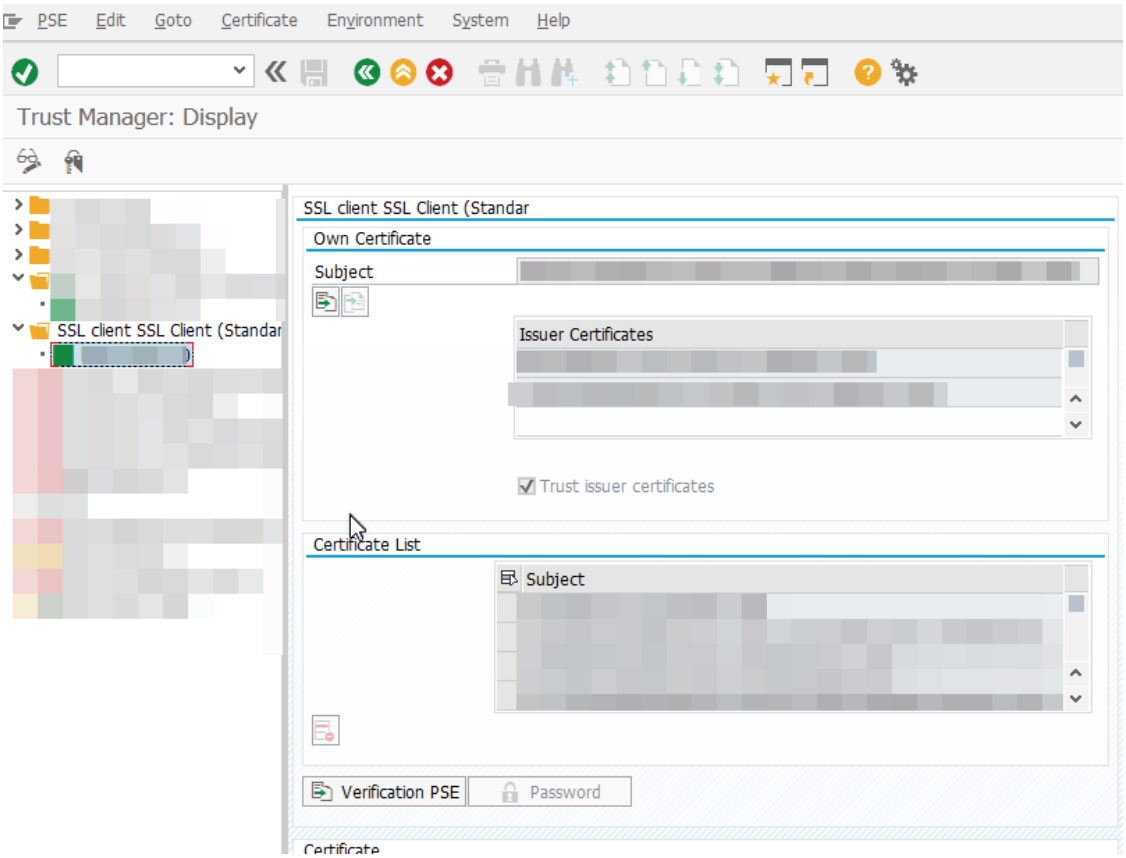


e. In the *Security Options* section of the *Logon & Security* tab, ensure that the `<SSL Secure Protocol>` is set to *Active*.

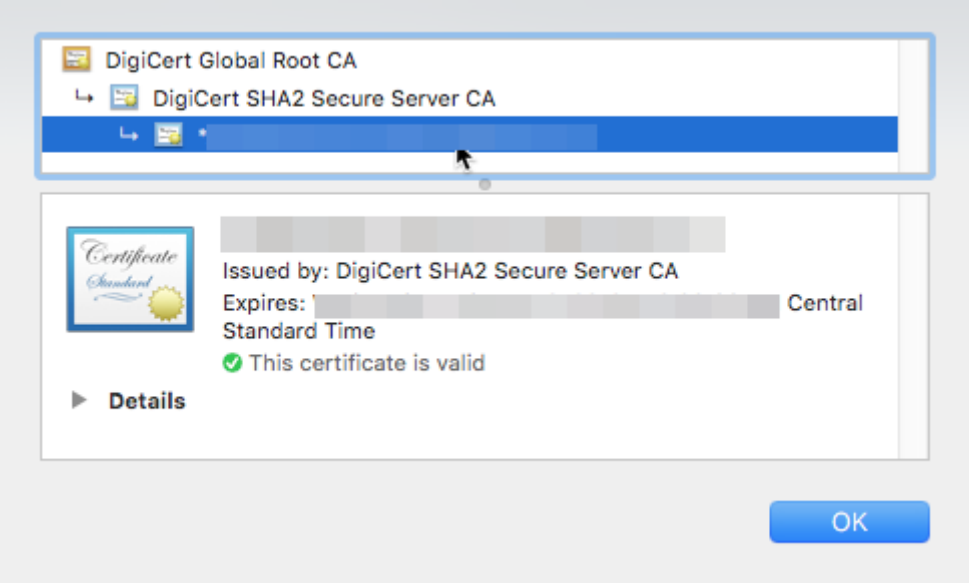
f. Remaining in the *Security Options* section, ensure that the *SSL Certificate List* used contains the SAP Business Technology Platform certificate chain and is active.



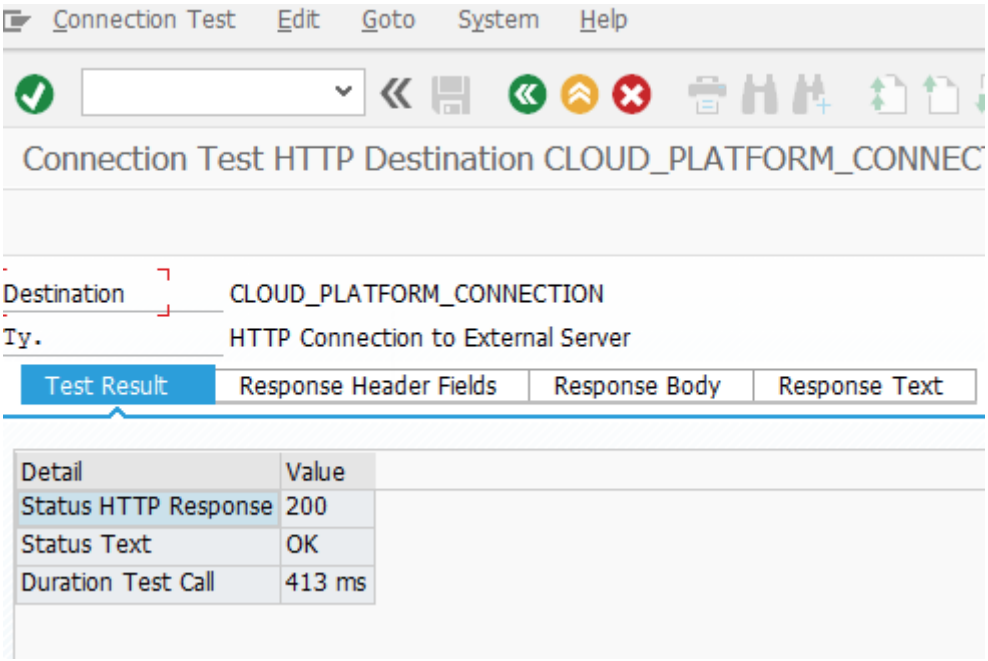
Check the certificate list of the SSL client by using the *STRUST* transaction. Check the matching trust list and ensure that the SAP BTP services certificate chains are in it.



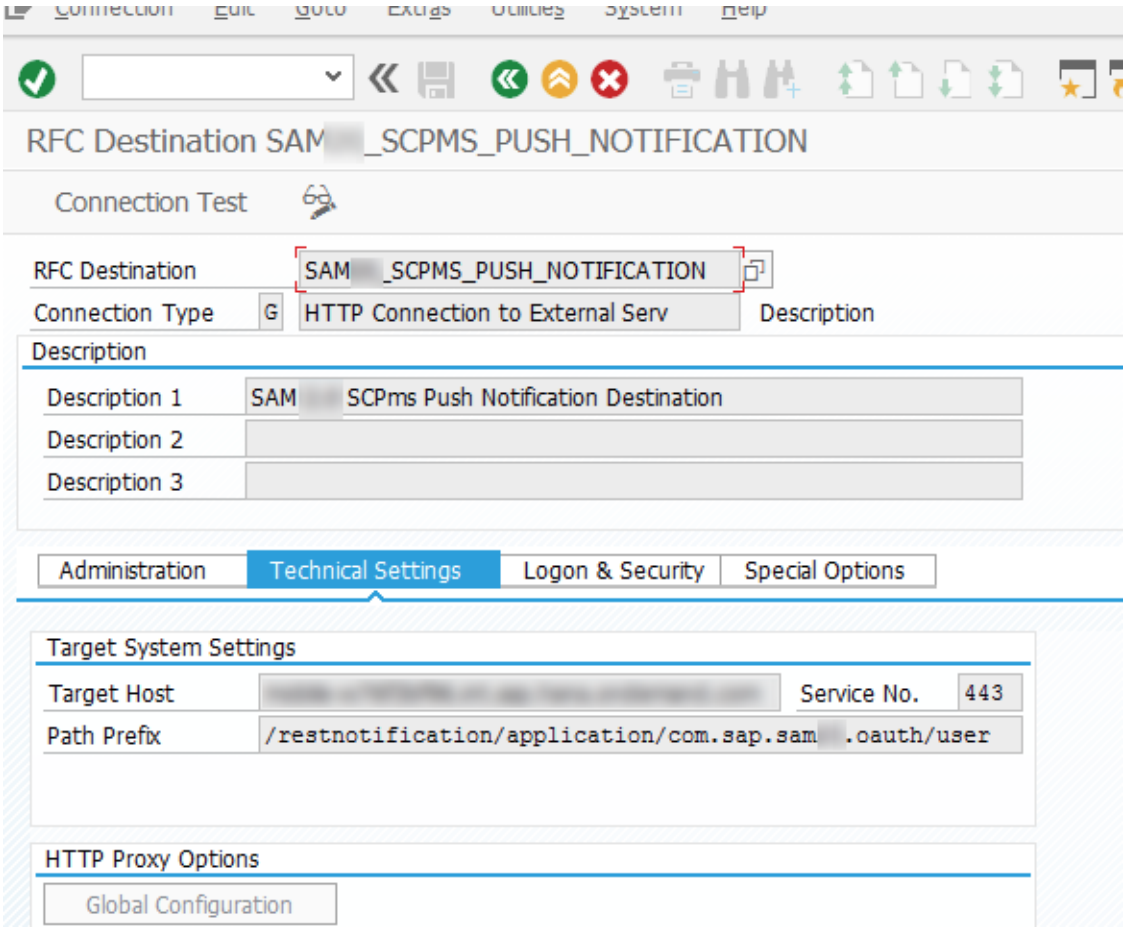
View the SAP BTP services certificate chain in any web browser by opening up the SAP BTP services cockpit and checking the security settings.



g. Save the connection and perform a connection test. If the configuration is completed properly, a 200 HTTP response is returned.

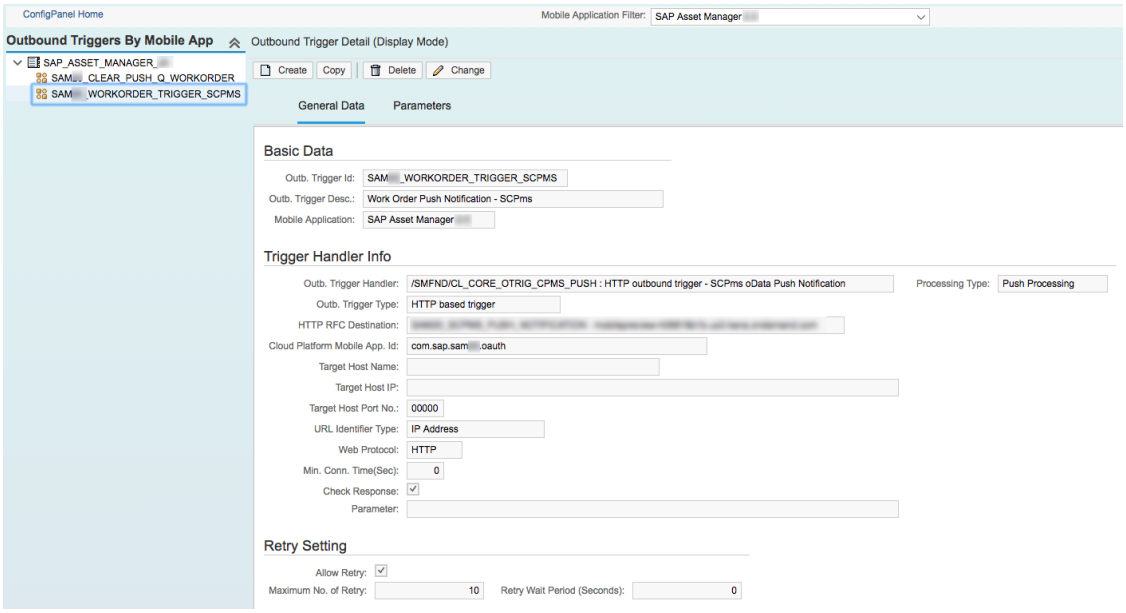


h. If the previous step correctly returned a status of *200*, add the rest of the path for the API to allow the POST to work properly.

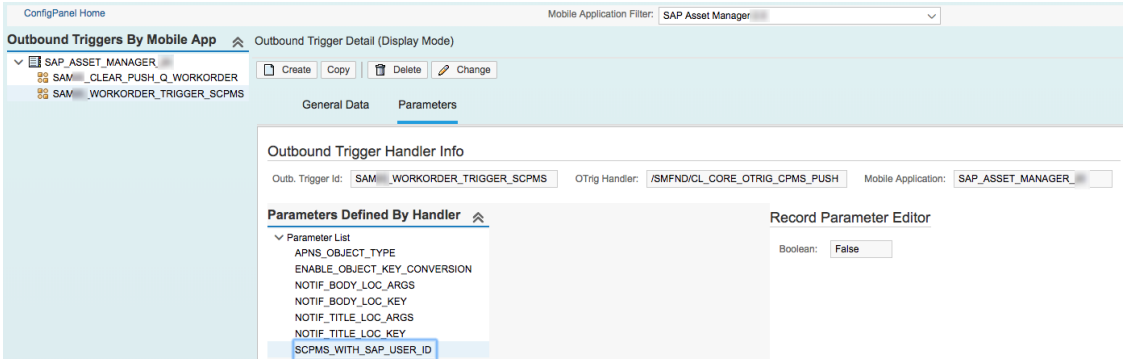


The back-end system can now send push notifications.

3. **Configure the back end to enable push from the SAP Service and Asset Manager application:**
   a. Using the ConfigPanel, navigate to the *Outbound Trigger Configuration* panel, *General Data* tab. Check the outbound triggers to ensure that the previously established *HTTP RFC Destination* is set as the RFC destination to the APPID that provides the push notifications to the mobile devices.



   b. Click the *Parameters* tab. If your user store on the back end and user store on the SAP Business Technology Platform are identical, set the *SCPMS_WITH_SAP_USER_ID* parameter value to *True*. If the user stores are not identical, set the parameter to *False*.



   c. Return to the ConfigPanel home screen and click the *Push Scenario Definition* link. Navigate to the *Outbound Trigger* tab. Find and highlight the push on the list of *Push Scenarios by Mobile App* and ensure that the outbound trigger is active.

Results

Push services are activated for SAP Service and Asset Manager. Thoroughly test the push functionality before deploying to the client devices.

3.5.6 Activating Default Push Services for SAP Service and Asset Manager - Cloud Foundry

Prerequisites

> **Note**
>
> The *SAP Service and Asset Manager Installation Guide* is a guide to setting up the basic framework necessary for push services using the default settings. For more details regarding configuration of push services, see the topic.

Before performing the procedure, ensure the following:

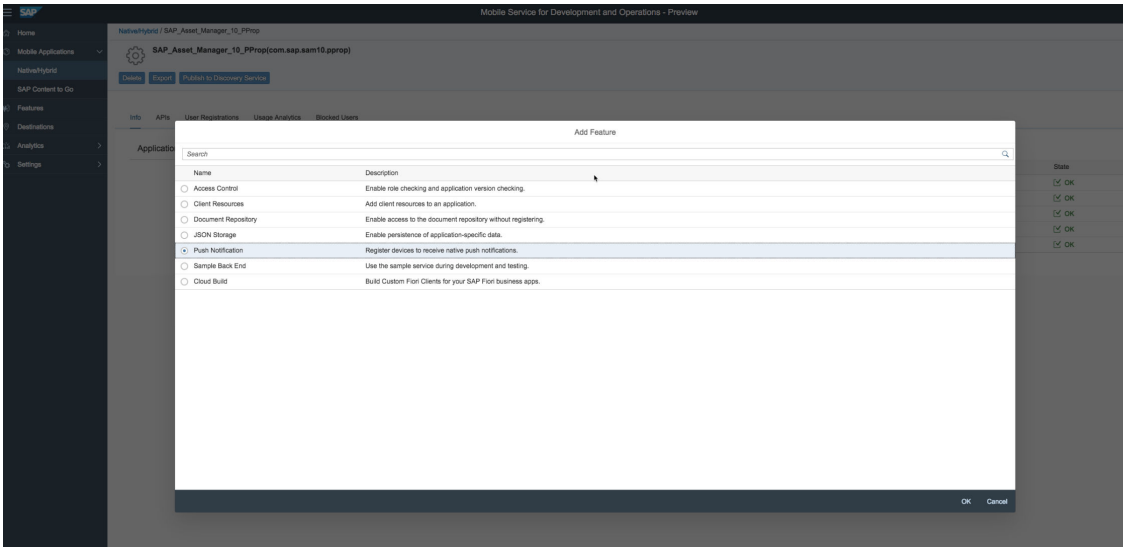
- SAP Service and Asset Manager application on the device is running on Mobile Development Kit 2.2.001
- You have installed SAP Service and Asset Manager 2310
- You have installed either SAP Mobile Add-On ECC or SAP Mobile Add-On for SAP S/4HANA. See the following installation guides on the following portal pages for version information:
  - SAP S/4HANA Mobile Add-On
  - SAP Mobile Add-On ECC

Context

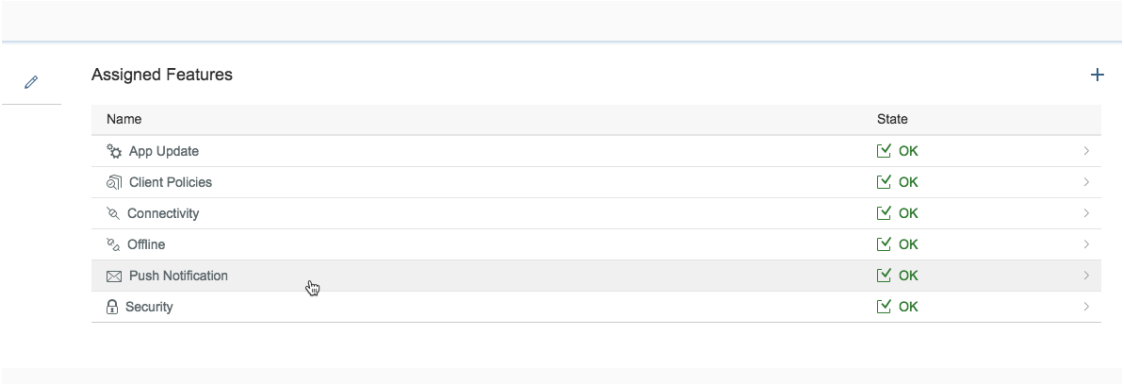
To configure and activate push for the Android platform, see the Configuring Push for Android [page 234] procedure.

Procedure

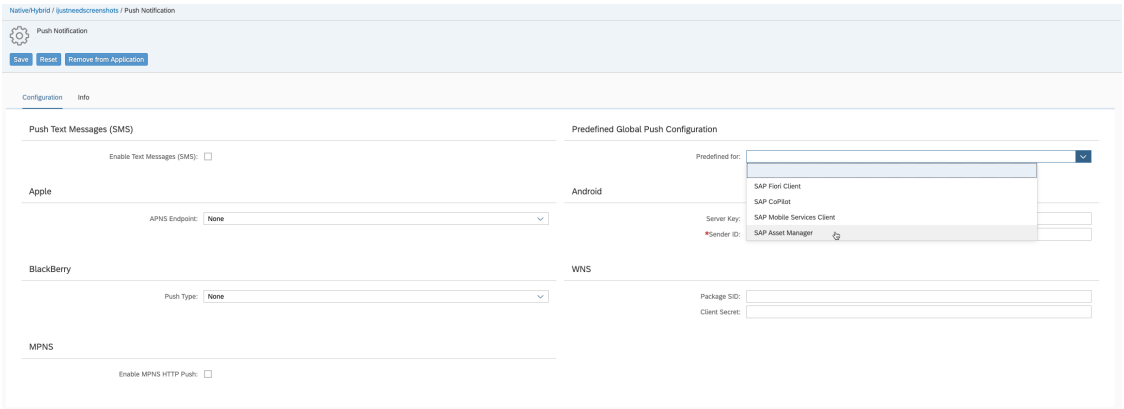
1. **Configure the SAP Business Technology Platform Mobile Services push API:**
   a. Enable the *Push Notification* feature in SAP BTP services:



b. Navigate to the *Push Notification* in the mobile app and create a new HTTP connection to your external server named *Assigned Features*:



c. If you are using a custom deployment of SAP Service and Asset Manager, upload the corresponding APNs certificates here. If you are using the default application provided by the Apple App or the Google Play store, select *Predefined for* SAP Service and Asset Manager in the *Predefined Global Push Configuration* section.

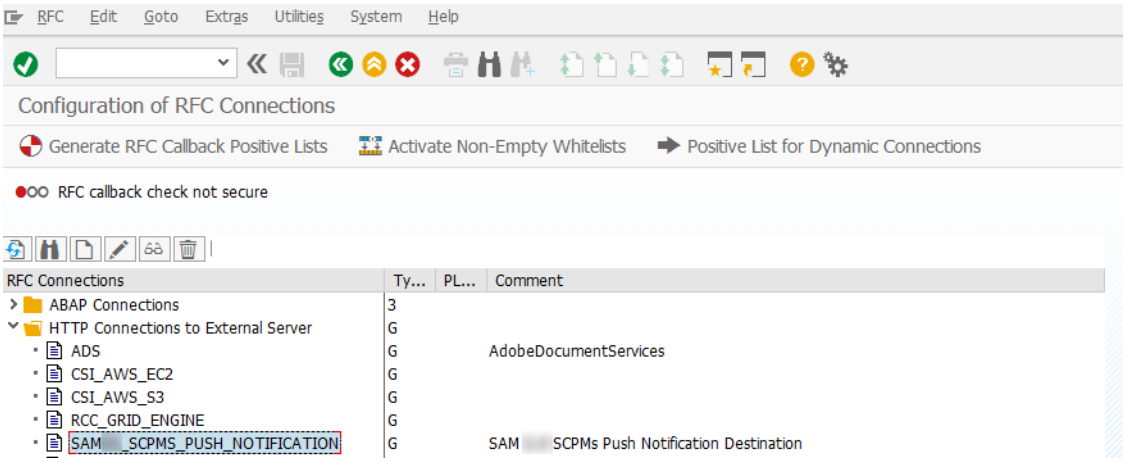


The SAP Business Technology Platform Mobile Services is configured for push.

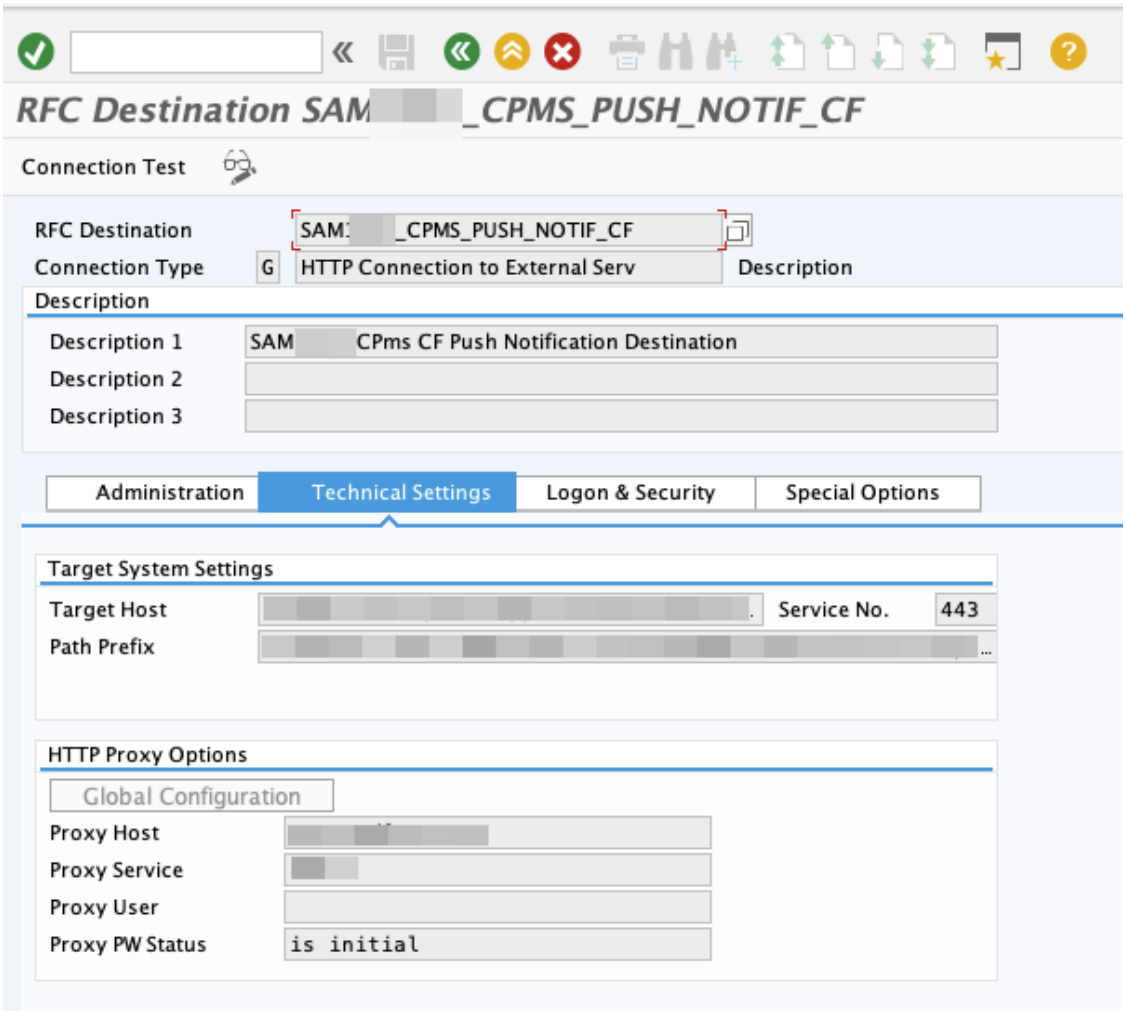
2. **Configure the back-end system to utilize the SAP BTP services push APIs:**

   a. Using the SAP GUI, run transaction *SM59*. Create a new HTTP connection with the name *SAM2310_SCPMS_PUSH_NOTIFICATION*.

> ### Note
>
> In a Cloud Foundry environment, you must use email instead of an I-number to utilize push notifications in the Administration portal.



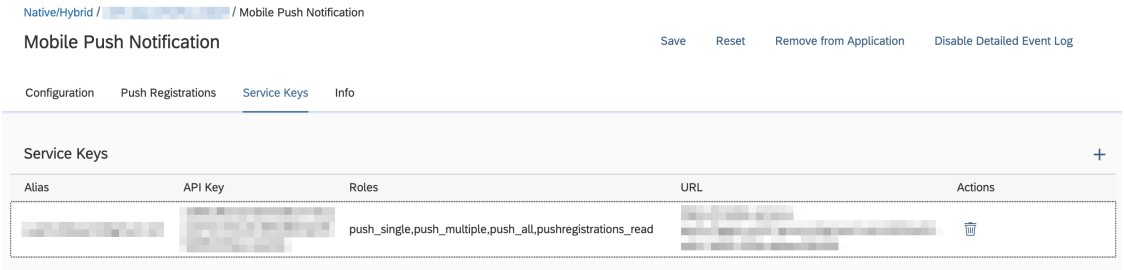
b. In the *Technical Settings* and tab of the new connection, set the *Target Host* to match the push API of the SAP BTP services, using *443* (the port number for HTTPS connections). Ensure the *Path Prefix* equals `/<Push URL GUID>/mobileservices/push/v1/backend/applications/<App ID>/notifications/users`



> **Note**
>
> If necessary, configure the proxy you are using to allow your back end to connect to the outside internet.

c. Click the *Logon & Security* tab. Under the *Logon Procedure*, select *Basic Authentication*. Enter the *Mobile Push Notification Alias* as the user name and the *Mobile Push Notification API Key* as the password.

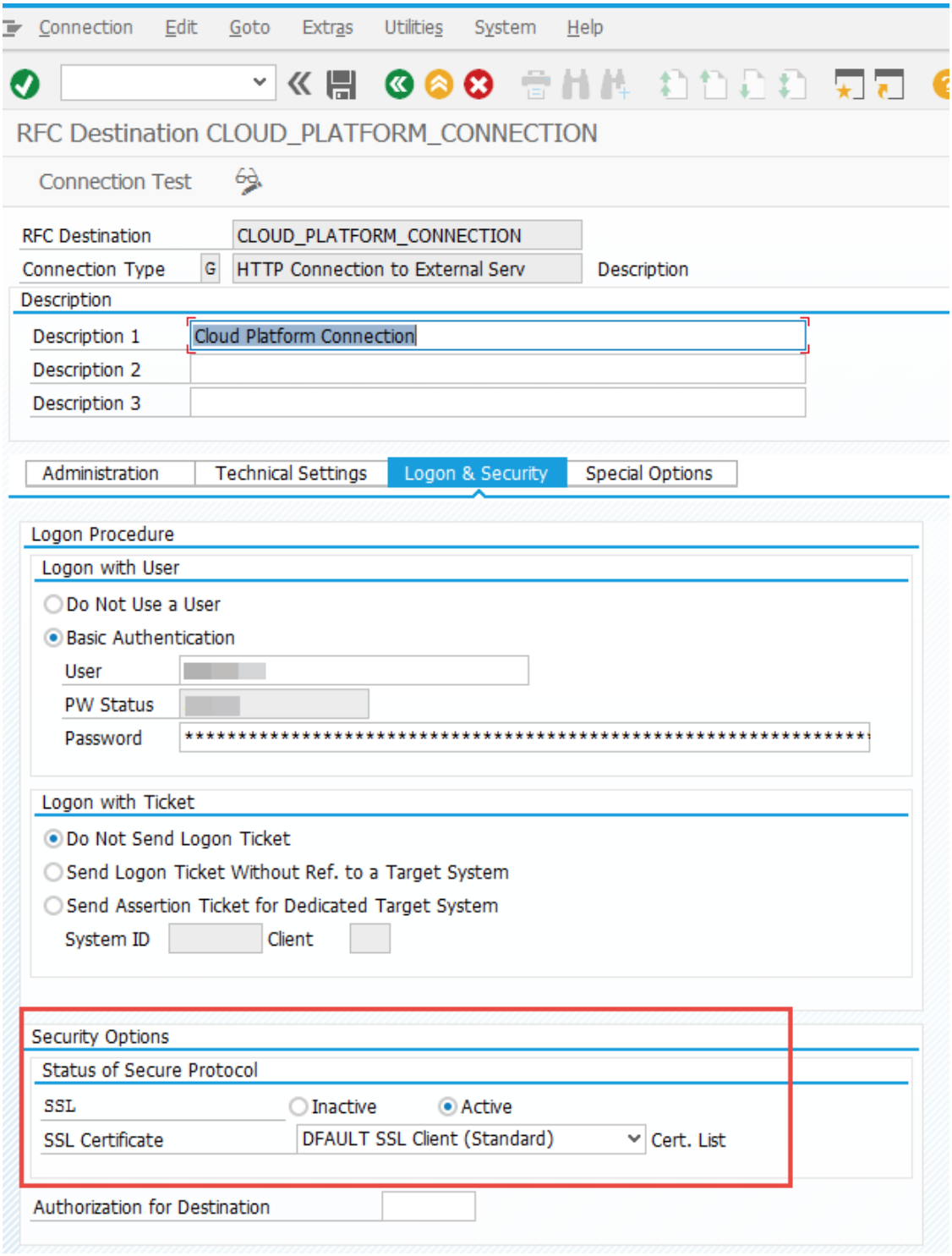
d. In *Mobile Services*, ensure your service user has the *API Key* assigned to them to ensure that the user is allowed to utilize the SAP BTP services API. This API key is the same value used as the password in the *RFC Destination*.



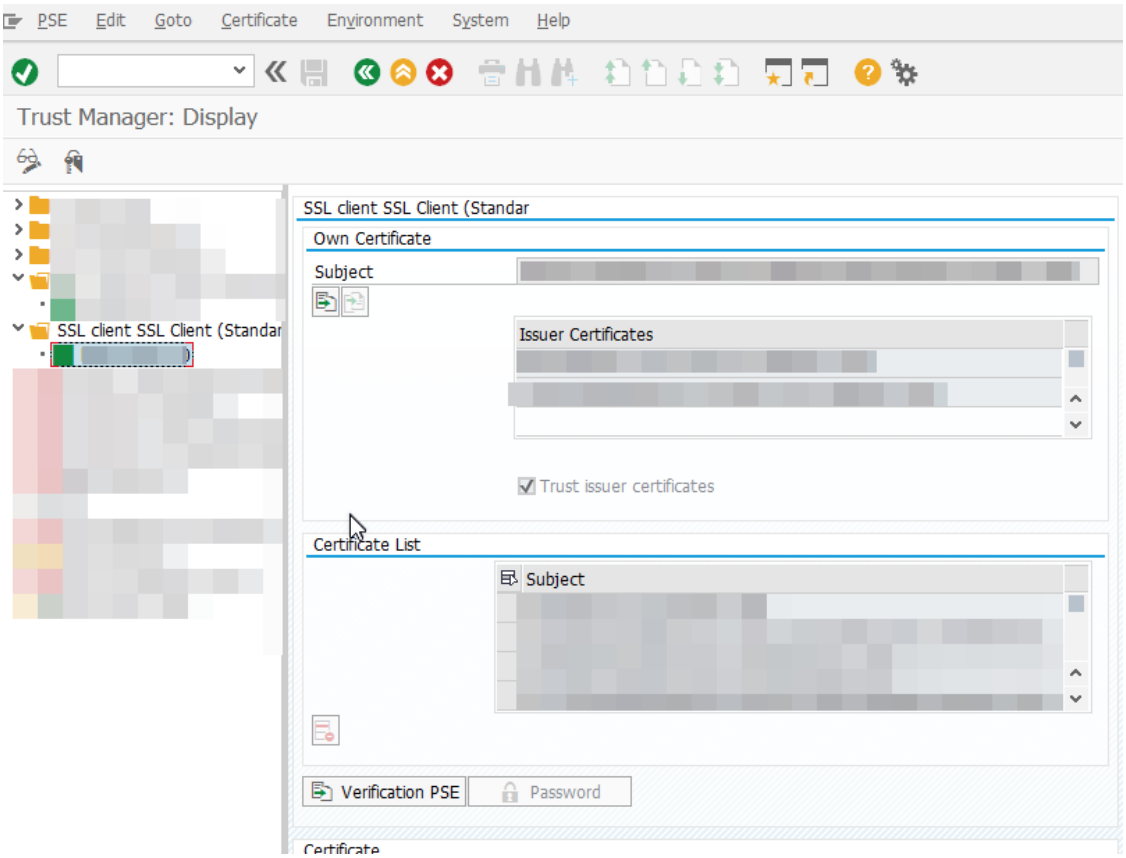
Basic authentication for this service user allows for free communication flow.

e. In the *Security Options* section of the *Logon & Security* tab, ensure that the `<SSL Secure Protocol>` is set to *Active*.

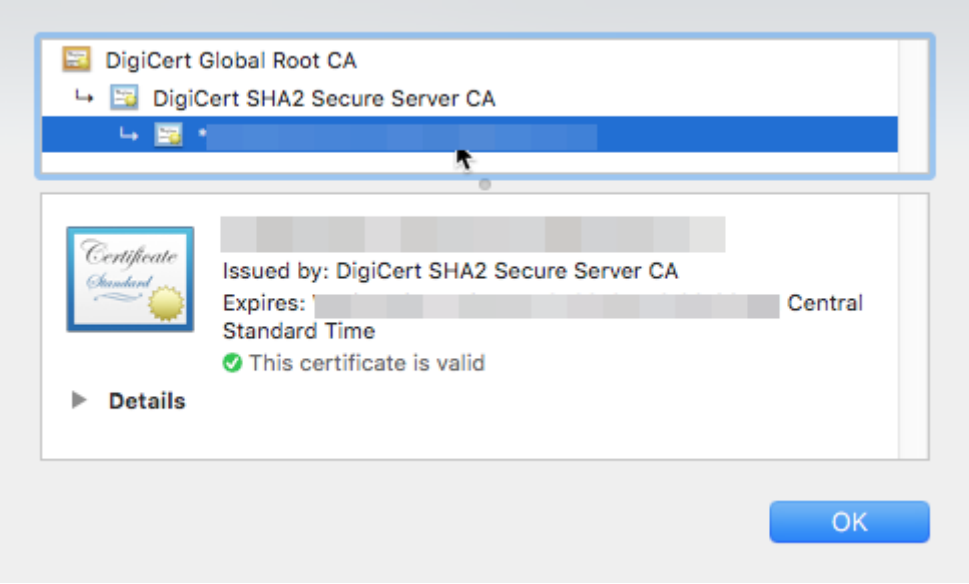
f. Remaining in the *Security Options* section, ensure that the *SSL Certificate List* used contains the SAP Business Technology Platform certificate chain and is active.



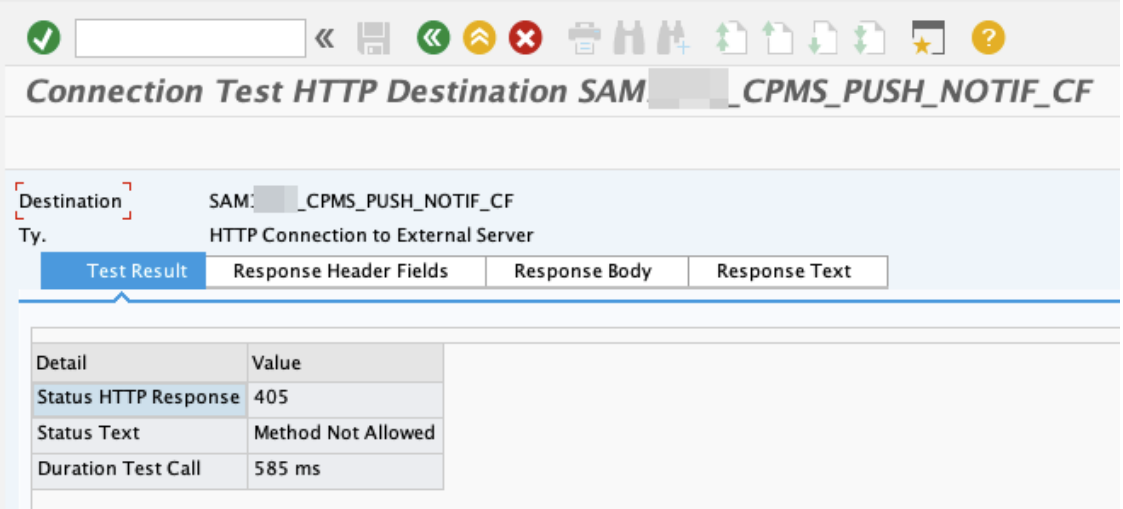
Check the certificate list of the SSL client by using the *STRUST* transaction. Check the matching trust list and ensure that the SAP BTP services certificate chains are in it.



View the SAP BTP services certificate chain in any web browser by opening up the SAP BTP services cockpit and checking the security settings.

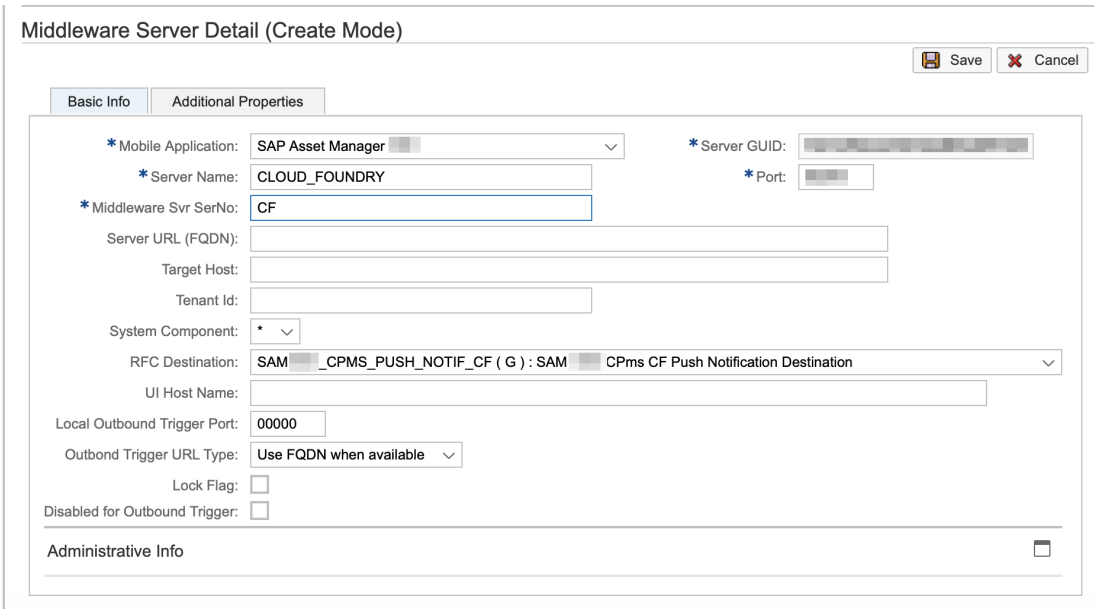


g. Save the connection and perform a connection test. If the configuration is completed properly, a 405 code is returned.



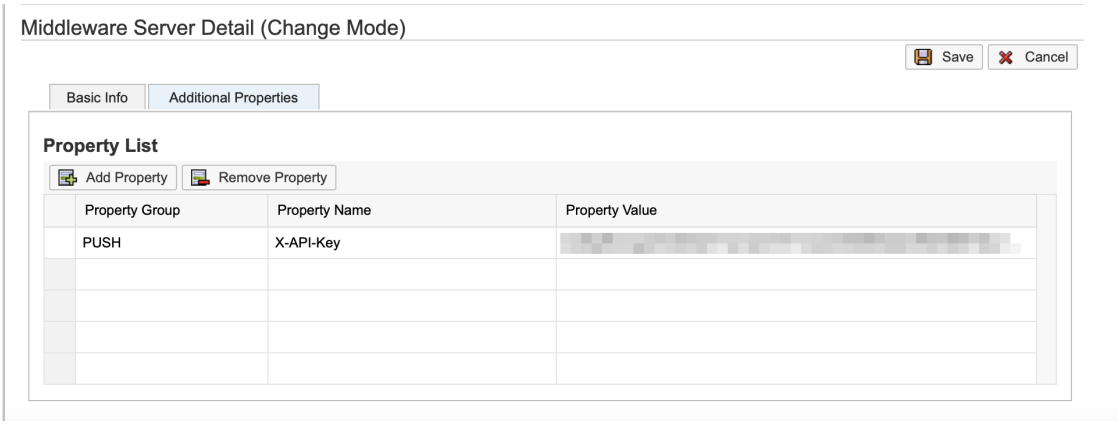
h. In the SAP GUI, open the Administration Panel using transaction */n/syclo/admin*.

i. Navigate to *Administration* > *Server Management*. Create a new middleware server, specifying the RFC destination you just created.



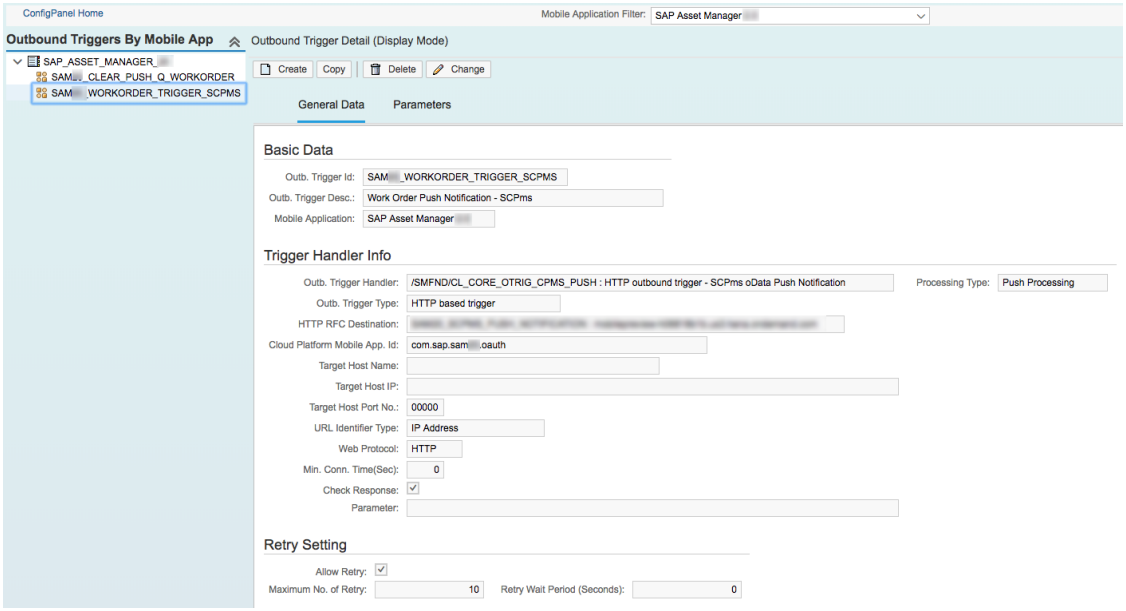
j. On the additional properties tab, create a new property with the following attributes:

- Property Group: PUSH
- Property Name: X-API-Key
- Property Value: Mobile push API key from SAP BTP services

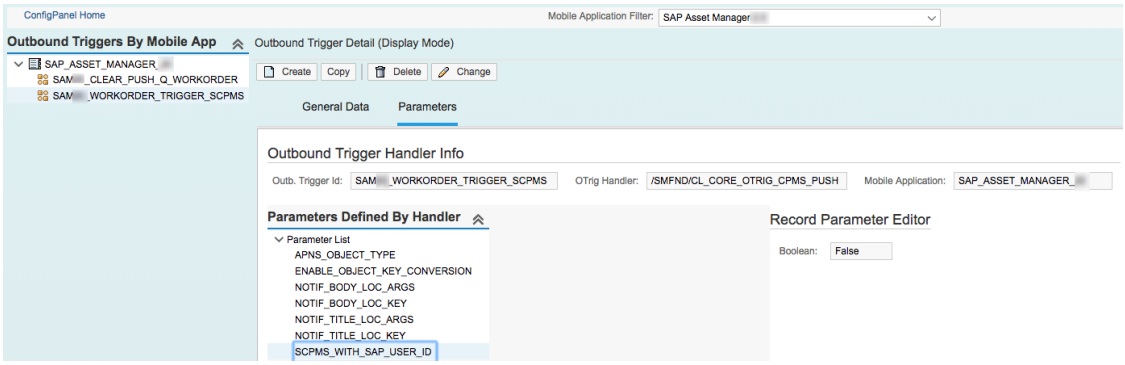


The back-end system can now send push notifications.

3. **Configure the back end to enable push from the SAP Service and Asset Manager application:**
   a. Using the ConfigPanel, navigate to the *Outbound Trigger Configuration* panel, *General Data* tab. Check the outbound triggers to ensure that the previously established *HTTP RFC Destination* is set as the RFC destination to the APPID that provides the push notifications to the mobile devices.



   b. Click the *Parameters* tab. If your user store on the back end and user store on the SAP Business Technology Platform are identical, set the *SCPMS_WITH_SAP_USER_ID* parameter value to *True*. If the user stores are not identical, set the parameter to *False*.



c. Return to the ConfigPanel home screen and click the *Push Scenario Definition* link. Navigate to the *Outbound Trigger* tab. Find and highlight the push on the list of *Push Scenarios by Mobile App* and ensure that the outbound trigger is active.

Results

Push services are activated for SAP Service and Asset Manager. Thoroughly test the push functionality before deploying to the client devices.

3.5.7 Configuring Push for Android

Firebase Cloud Messaging (FCM) is a cross-platform cloud solution for messages and notifications for Android, iOS, and web applications.

Context

To enable push notification for the SAP Service and Asset Manager application using the Android platform, use the following procedure:

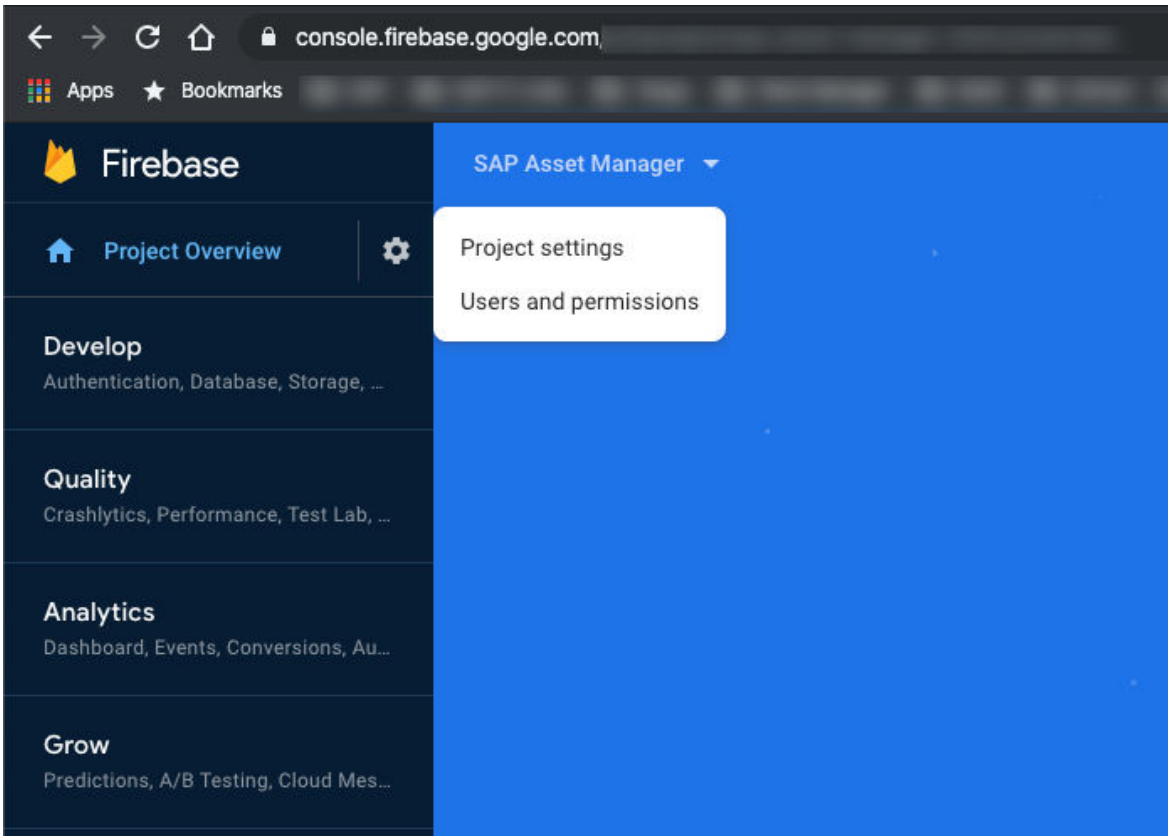
Procedure

1. Create a free Firebase account. See the main Firebase page to set up a new account, or connect an existing account.

   > **Note**
   >
   > Documentation on Firebase is found on the Firebase Documentation page.

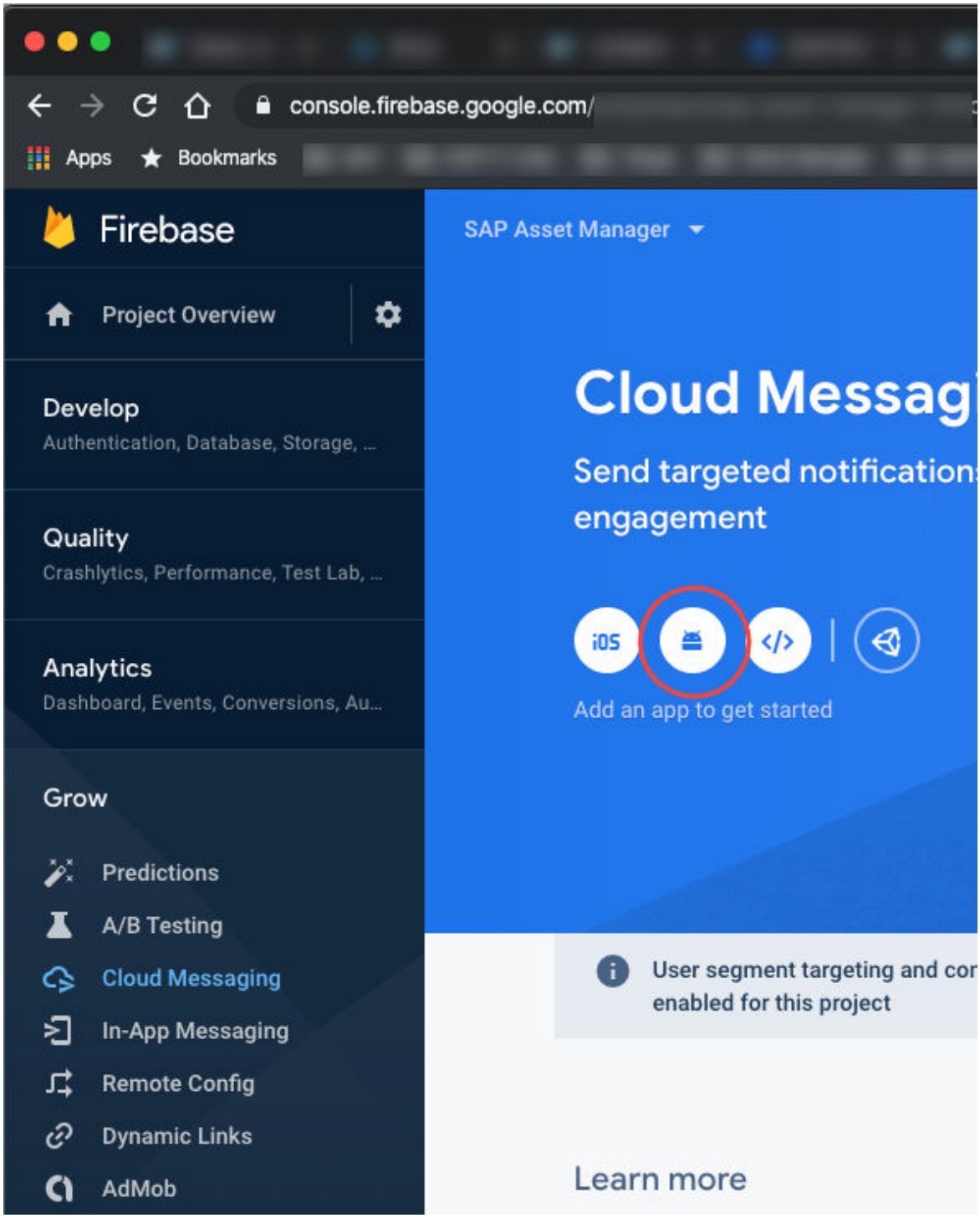
2. Add a new project, or use an already-existing project.

3. Click the gear icon and select the *Project Settings* menu option. Click the *Cloud Messaging* tab.



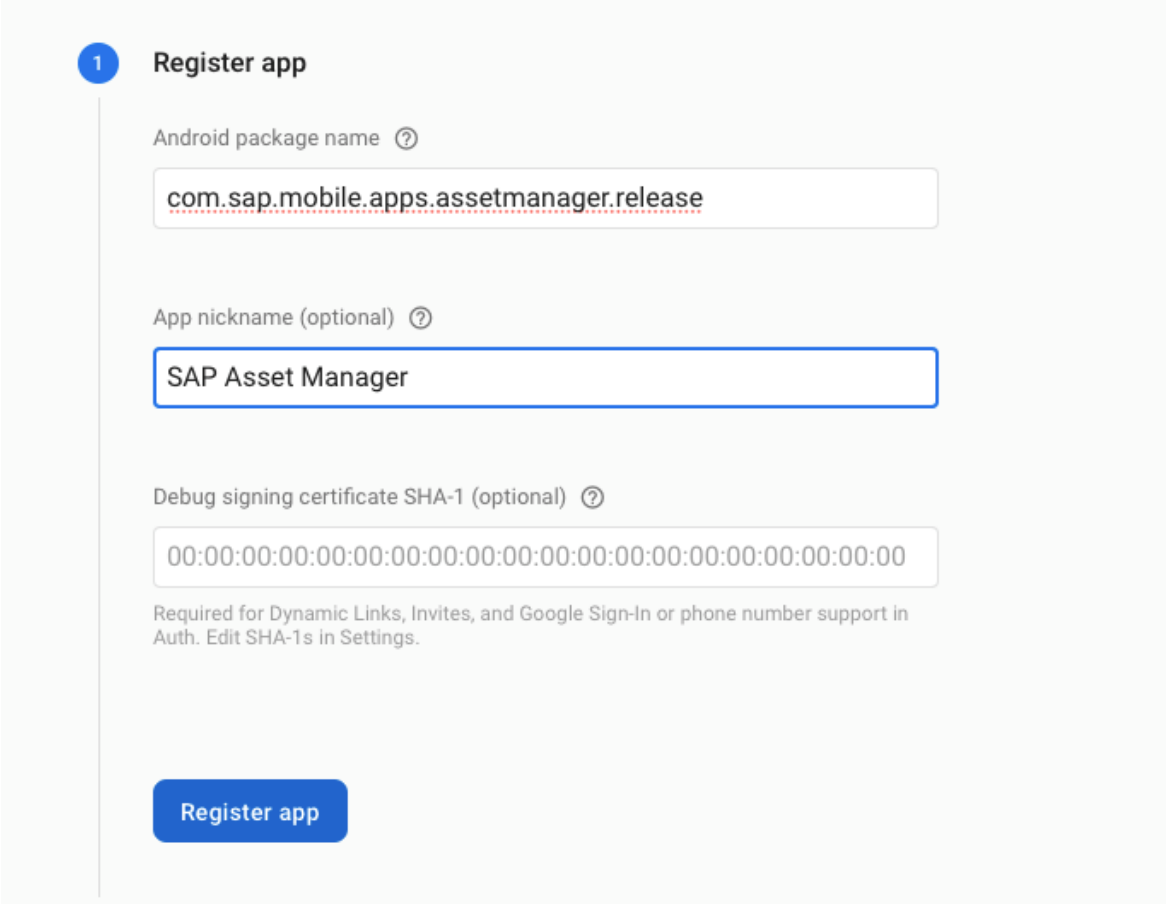
4. Copy and paste the information in the *Server Key* field to use in a later step.
5. Navigate to *Grow* > *Cloud Messaging* using the sidebar.
6. Click the Android icon to add a new Android app.

   The *Add Firebase to your Android app* window appears.



7. Add your package name to the *Android package name* field in *Step 1*. Ensure that the package name matches the *BundleID* found in the `SAM.mdkproject/MDKProject.json` file. When done, click *Register app*.

   Your app is registered and you're moved to *Step 2 - Download config file*.



8. Click the *Download google-services.json* button located in *Step 2 - Download config file*.

   The `google-services.json` file is downloaded to your computer.
9. Navigate to `SAM.mdkproject/App_Resources/Android`. Paste the `google-services.json` file you downloaded in the previous step to this location.
10. Build the SAP Service and Asset Manager client. For detailed instructions on building the client, see SAP Asset Manager Installation Guide, *Building / Branding the SAP Asset Manager Application* chapter.
11. Copy and paste the server key you saved in *Step 3* inside your SAP BTP services push notification configuration in the *Android* section.

    For instructions on how to enable Android push notifications in SAP BTP services, see the Android Push Notifications procedure.

    For detailed information on configuring push for the SAP Service and Asset Manager application, see the procedure Activating Default Push Services for SAP Asset Manager, specifically the screenshot in *Step 1d*.

Next Steps

Continue to the procedure Setting up the Outbound Trigger for your Push Configuration [page 214].

Related Information

https://help.sap.com/viewer/DRAFT/f15c174c3c3647088d38fb220e42c006/latest/en-US/5629175b49304549b724bc838c9a72ec.html

4 Setting up an OData Mobile Data Object

For OData troubleshooting information, see OData API in the *SAP Cloud Integration* documentation.

4.1 Supported OData Features for SAP Mobile Add-On

The following table lists the OData features that SAP Mobile Add-On supports.

OData Feature	Status
Create	Maps to HTTP POST method
Read	Maps to HTTP GET method
Update	Maps to HTTP PUT method
Delete	Maps to HTTP DELETE method
Query	Maps to HTTP GET method
$select	Specifies a subset of properties to return
$top	Determines the maximum number of records to return
$skip	Sets the number of records to skip before it retrieves records in a collection
$filter	Specifies an expression or function that must evaluate to *true* for a record to return to the collection
$count	Returns the number of records in a collection
$orderby	Determines which values are used to order a collection of records
$expand	Specifies that related records must be retrieved in line with the record or collection being retrieved. For example, use *$expand* to retrieve a customer and all orders placed by that customer in a single query.
$inlinecount	Supported
$skiptoken	Supported

OData Feature	Status
$format	Supported
Navigation	Supported
Delta token	Supported
Tombstone	Supported
Complex types	Supported
$batch	Supported
Deep insert	Supported via single post operation and through $batch request using content ID referencing
Custom query options	Not supported
$link	Not supported
$value (media links or attachments)	Supported
ETags / concurrency control	Supported
$filter Details	

String functions
• Supported:
  • bool substringof(string p0, string p1)
• Not Supported:
  • string trim(string p0)
  • string concat(string p0, string p1)
  • int length(string p0)
  • int indexof(string p0, string p1)
  • string replace(string p0, string find, string replace)
  • bool endswith(string p0, string p1)
  • bool startswith(string p0, string p1)
  • string toupper(string p0)
  • string substring(string p0, int pos)
  • string substring(string p0, int pos, int length)
  • string tolower(string p0)
	Partially supported

Date functions	Not supported
Math functions	Not supported
Arithmetic operators	Not supported

OData Feature	Status
Type functions	Not supported
$filter supported on complex type properties within an entity	

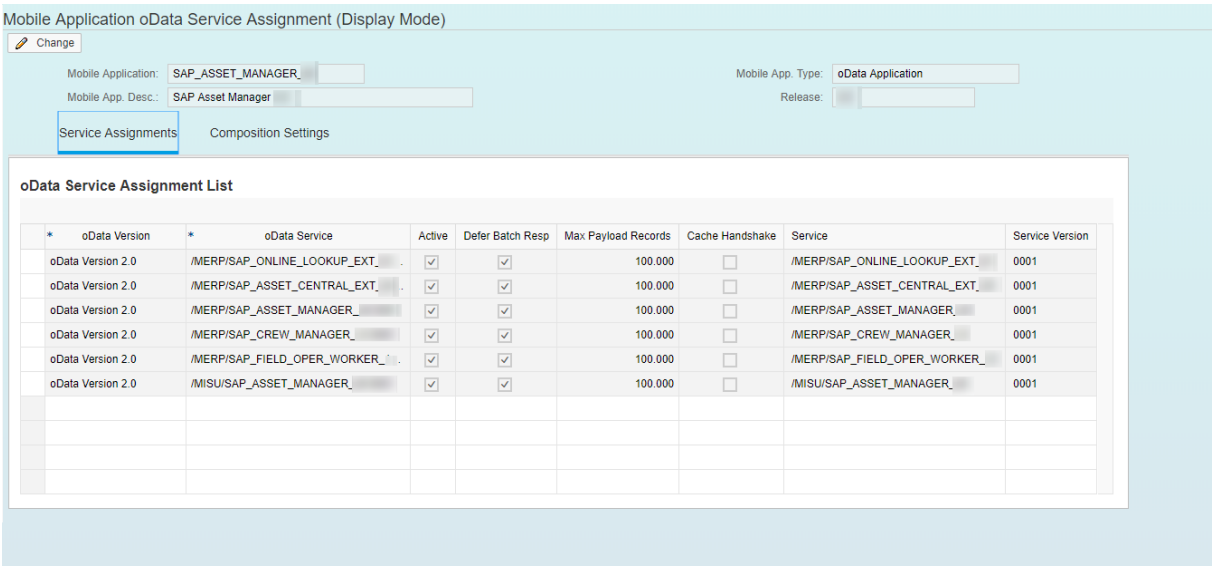
> **Note**
>
> For related constraints, see SAP Note 1830712.

4.2 Setting the OData Mobile Data Object Service Assignment

You can assign SAP system aliases to a service. With the assignment, an OData request from an SAP Gateway consumer can be routed to the corresponding back end service.

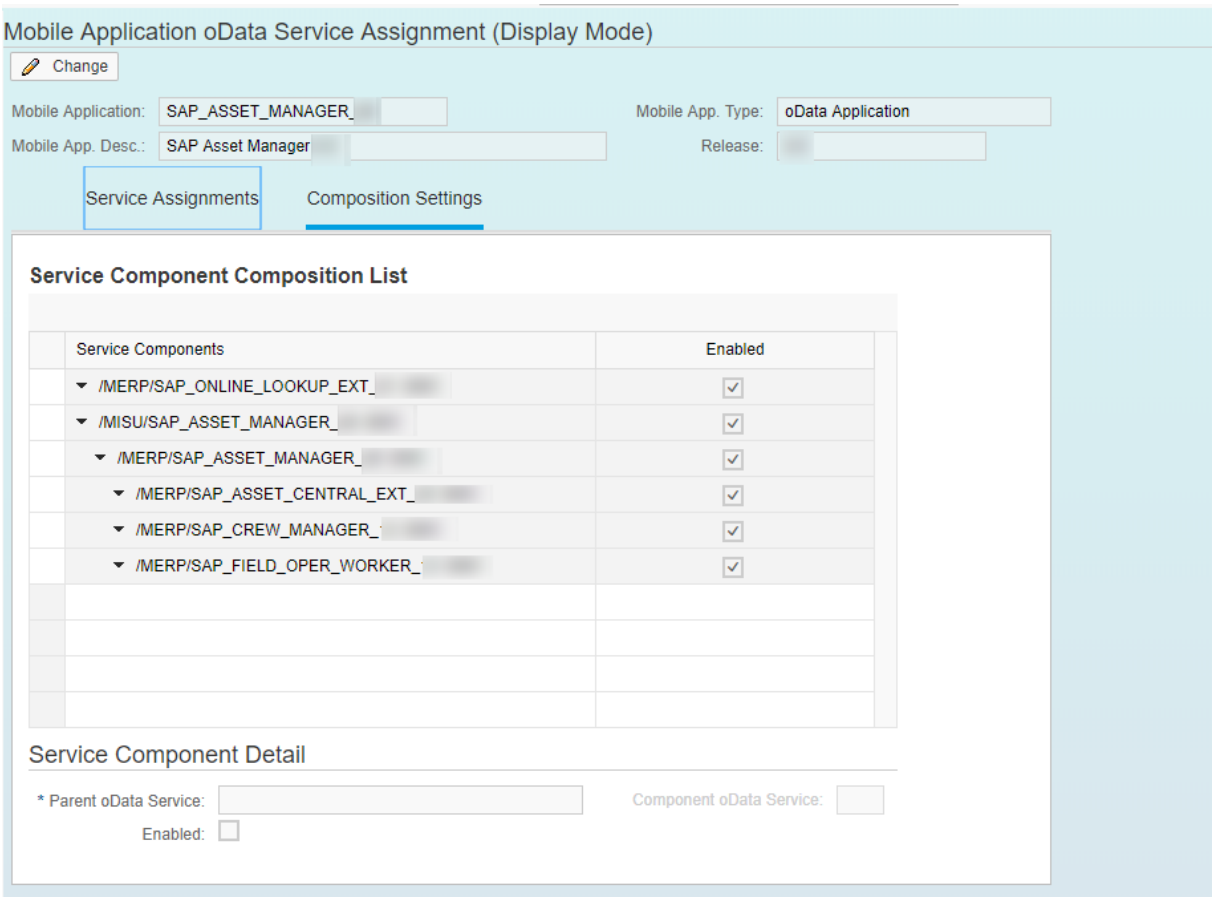
Context

Assign OData services to the SAP Asset Manager application using the *Service Assignments* tab.



Build a hierarchy between assigned services using *Composition Settings*. To utilize OData entities from a different service such as the Crew Management and Field Operations Worker component service, add the relevant OData services (*/MERP/SAP_CREW_MANAGER_<XX>* and */MERP/*

SAP_FIELD_OPER_WORKER_<XX>) under the SAP Service and Asset Manager service as shown in the following example:



Procedure

1. Ensure that your mobile application is selected in the *Mobile Application Filter* field at the top of the page.
2. Expand the *Mobile Application List* in the left pane and select your mobile object.

   Your chosen mobile application OData service assignment details are displayed in the main window on the *Service Assignments* tab.
3. Click the *Change* button to change the existing mobile service assignment details or to add a new mobile service assignment.
4. To add a new mobile service assignment, click the *Assign OData Service* button.
   a. Select an *OData Version*, if there is more than one to choose from, from the dropdown menu.
   a. Select an *OData Service*, or system alias, from the dropdown menu.

      The corresponding back-end server is populated in the *Service* field.
   b. To *Defer Batch Response*, mark the checkbox. That is, you are setting the OData service to process all of the CHANGESET operations at once (deferred processing).
   c. To activate your new service assignment, check the *Active* checkbox.

5. On the *Composition Settings* tab, build a hierarchy between your OData services to include all metadata from different child services. You can add and remove component assignments depending on your service needs.
6. To add a component assignment, choose *Add Component Assignment*.
   a. Select a *Parent OData Service* from the dropdown list.
   b. Select *Component OData Service* from the available list to make it a child service for the parent OData service.
   c. To activate the new component assignment, check the *Enabled* checkbox.
7. To remove the component assignment, select the service component you want to remove and choose *Remove Component Assignment*.

   The hierarchy relationship with its parent service component is removed.

Next Steps

Perform the following procedures:

- Setting the OData Mobile Data Object Configuration [page 243]
- Setting the OData Model Configuration [page 245]

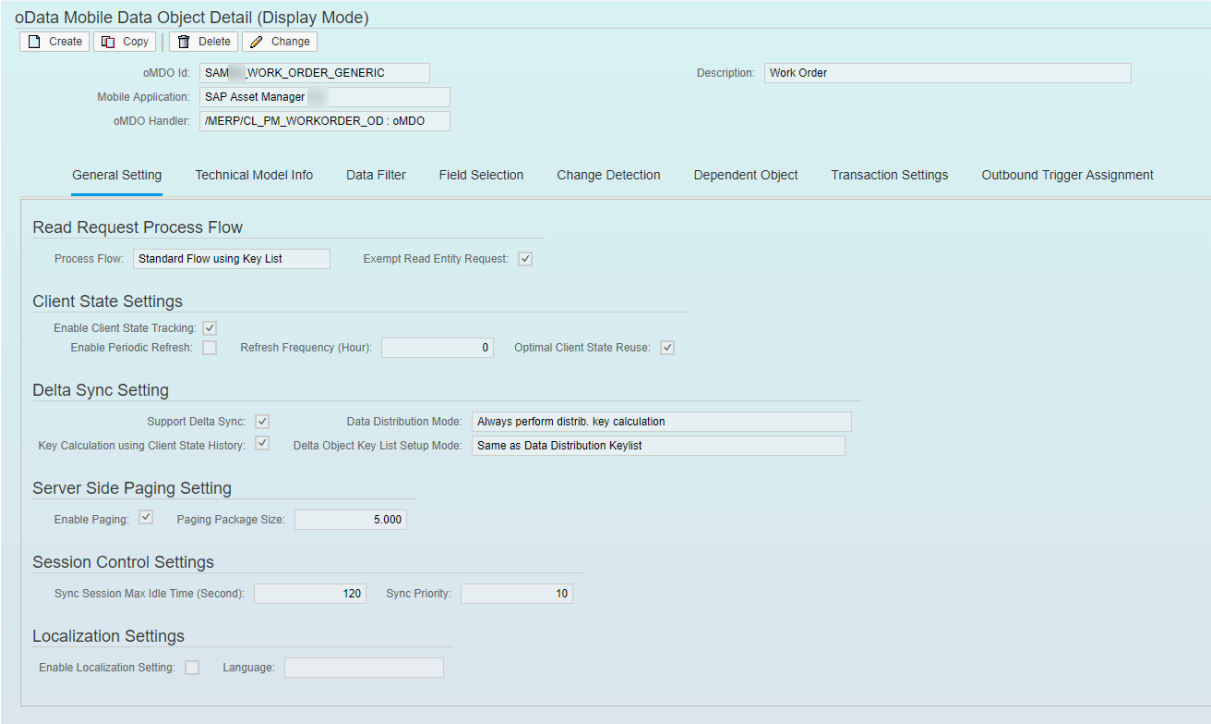
4.3 Setting the OData Mobile Data Object Configuration

Prerequisites

If you are setting up a new OData mobile data object, or changing an OMDO, read and perform the following procedures before performing this procedure:

- Setting the OData Mobile Data Object Service Assignment [page 241]

Context



Procedure

1. Navigate to and click the *Mobile Application Integration Framework Configuration Home* > *OData Mobile Data Object Configuration* link.

   The *OData Mobile Data Object Configuration Detail* page displays.
2. Select your application in the *Mobile Application Filter* field. Selecting an application filters the OData Mobile Data Object by Mobile App choices in the left panel to only those available in your chosen application.
3. If you are copying an OMDO, which is recommended, choose your OMDO from the *OData Mobile Data Object by Mobile App* list on the left and copy it to your customer namespace. See the Copying an Object to the Customer Namespace [page 147] procedure for more details. Do not modify the original OMDO.
4. Enter an OMDO ID in the field with customer namespace.
5. Select a *Mobile Application* from the dropdown list.
6. Select an *OMDO Handler* from the dropdown menu. The OMDO handler is the name of the handler as defined in the system.

   The OMDO handler will provide the data source for the entity record.
7. Enter a short *Description* of your new OData mobile data object.
8. Choose one of two settings for the *Process Flow* in the *Read Request Process Flow* section:
   - **Standard Flow Using Key List**

- **Basic Flow without Key List**

9. You can keep the *Client State Settings* from the original OMDO, or change it depending on your business needs.
10. You can keep the *Delta Sync Settings* from the original OMDO, or change it depending on your business needs.
11. Check the *Enable Paging* checkbox to set the package size for the SAP BTP services. Type in a number for the default package size in the *Paging Package Size* field.

Next Steps

Continue to Setting the OData Model Configuration [page 245].

4.4 Setting the OData Model Configuration

An OData model gives detailed information about each object in an OData feed. You can define a new data model in your application to suit your requirements based on the data you want expose at runtime.

Prerequisites

Complete the following procedures:

- Setting the OData Mobile Data Object Service Assignment [page 241]
- Setting the OData Mobile Data Object Configuration [page 243]

Context

Entity Sets are used to group instances of an entity type together with instances of any type that are derived from this particular entity type. You can access the OData entity details from the ConfigPanel home page by choosing *OData Model Configuration*.

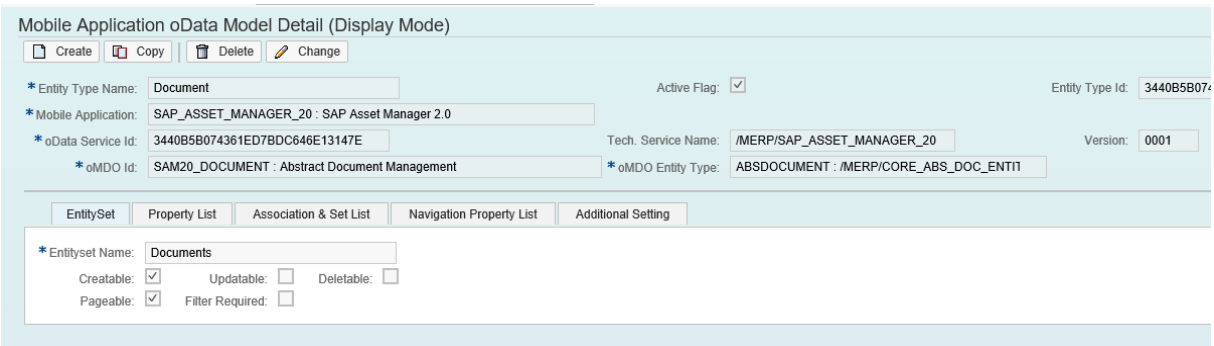
You can define properties for entity types on the **Property List** tab. Properties define the characteristics of data that an entity type instance contains at runtime.

An association is a named relationship between different entities. An association defines a peer-to-peer relationship between participating entity types, and it can also support different multiplicities at both ends. You can define associations between current and dependent entities on the **Association & Set List** tab. An association set groups association instances, and specifies the relationship between two entity sets based on the respective entity types of the underlying association.

Navigation properties describe the association relationship between two entities. The navigation property is tied to an association, and it allows the navigation from one end of the entity type, which declares the

navigation property, to the other related end. The entity types include one or more navigation properties and can be defined on the *Navigation Property List* tab.

Finally, you can set the bind structure conversion exits and the Media flag for entity type on the *Additional Setting* tab.



> **Note**
>
> Optional steps are included to explain the required fields when creating a new OData model. These fields are grayed out when you are working with a copied OData model and you can ignore them in the procedure.

Use the following steps as a guide to change an existing OData model:

Procedure

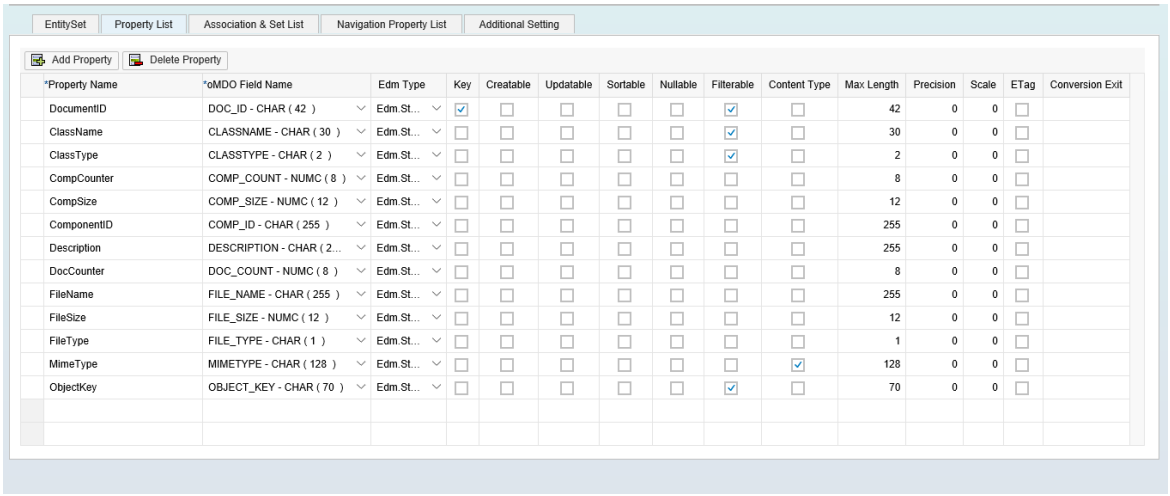
1. Navigate to and click the *Mobile Application Integration Framework Configuration* > *OData Model Configuration* link.

   The *Mobile Application OData Model Detail* page displays.
2. Select your application in the *Mobile Application Filter* field. Selecting an application filters the OData Model Entity Type by Mobile App choices in the left panel to only those available in your chosen application.
3. Once you have selected the mobile application, select *OData Service ID* from the drop-down list.

   Note that you cannot share models between OData services. Each service has its own model.
4. If you are creating a new OData model, click on *Create* button on the top and type an entity type name in the field. The entity type name represents the structure or a single record.
5. Select an *OMDO ID* from the drop-down list. The OMDO ID is the object that is providing the data for the record.
6. Select an *OMDO Entity Type* from the drop-down list. The OMDO entity type is the source that provides information to the OData model. When a service request for the entity type occurs, the OData model invokes the selected OMDO ID and the related handler method.
7. Type an *EntitySet Name* into the field. While an entity type describes a data structure, an entity set contains the instances of the given structure. Therefore, a best practice for an entityset name is to create a plural of an entity type name. For example, if an entity type name is *Test*, the entityset name will be *Tests*.
8. Check any of the following checkboxes to enable additional OData features. Note that some may require additional configuration on other tabs or links.
   - *Createable*: Similar to a POST request in REST

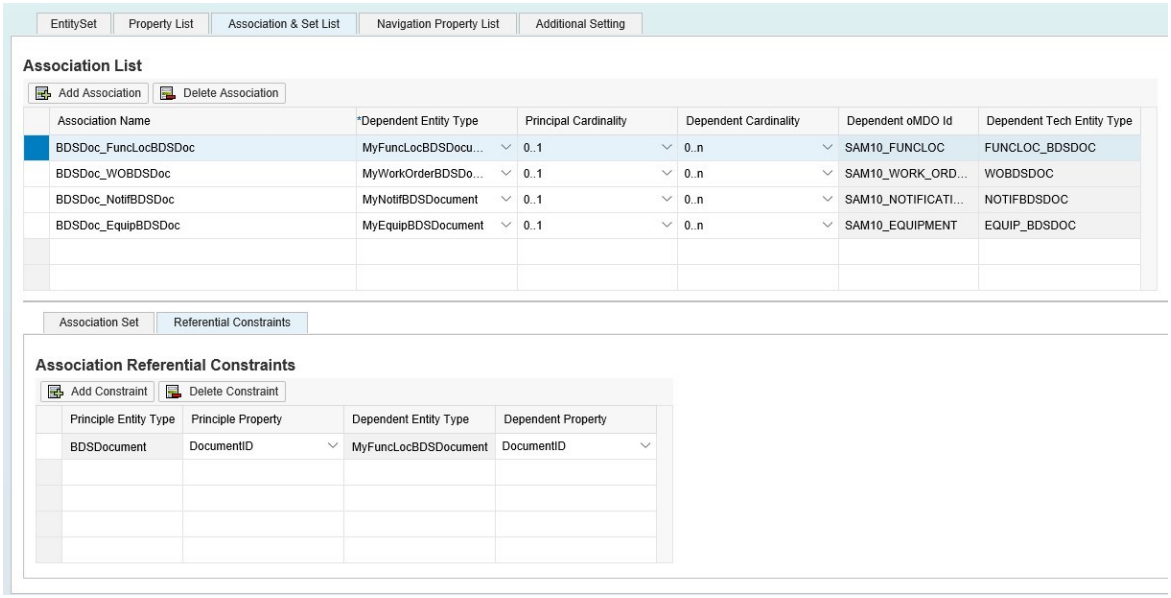
- *Pageable*: Enables data staging. See Setting the OData Mobile Data Object Configuration [page 243], *General Setting* tab, *Server Side Paging Setting* section, for more information.
- *Updateable*: Similar to a PUT or MERGE request in REST
- *Filter Required*: When checked, the entityset cannot be accessed directly. That is, the entityset is only valid within its containing entity and is not visible in the service document. You cannot query the entityset without using a `$filter` expression. If you try to access the entity set without using a `$filter` expression, a message displays that explains which filter expressions are required, at a minimum.
- *Deletable*: Similar to a DELETE request in REST

9. Click the *Property List* tab.



10. To add a new property to the entity type, click the *Add* button.
    a. Type the property name into the `<Property Name>` field.
    b. Select an *oMDO Field Name* from the dropdown list.
    c. Select the appropriate *EDM Type* (Entity Data Model) from the dropdown list.
    d. Check the *Key* column for Key fields.
    e. Define the attributes of the new property depending on the scope of the entity type.

    If you use the *Datetime* Edm Type and its related properties as an optional field, set the attribute *Nullable* to true.
11. Click the *Association & Set List* tab.

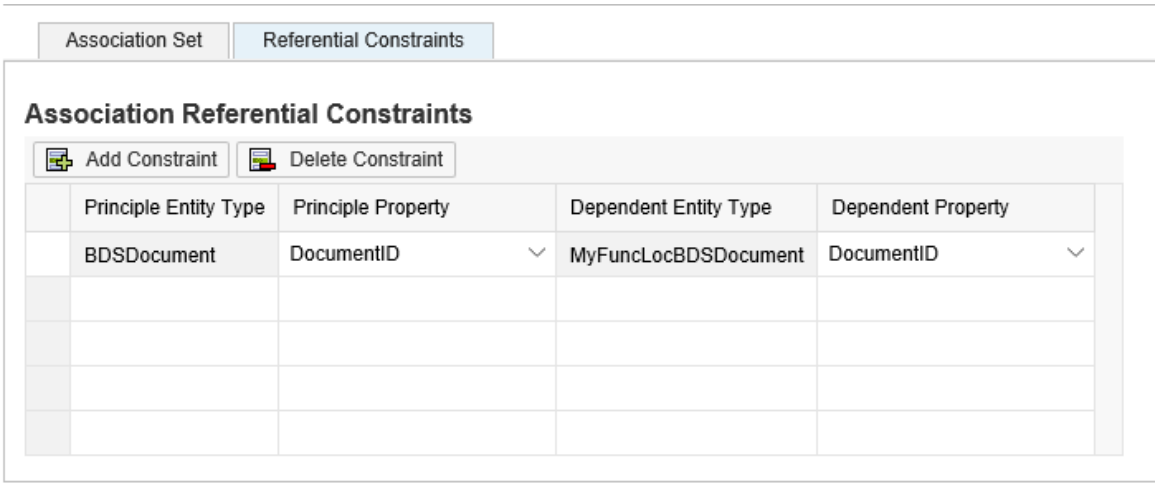


Define associations between entities to express relationships between entities.

Associations themselves are freestanding. Specify on top of the associations, which of the entities participating in the relationship can navigate over the association to the other entity using the *Referential Constraints* tab.

12. Click the *Add Association* button to add a new association. Associations define a peer-to-peer relationship between participating entity types, and can support different multiplicities at both ends.
    a. Type a name for your new association in the *Association Name* field.

       Your Association can be either internal or external when adding a new association; by default the current entity will be the principle entity. If you want to add an external association where the current entity is treated as dependent entity, select the *External Association* checkbox.
    b. Select the dependent entity from the *Dependent Entity Type* drop-down menu for internal association, whereas select the *Principle Entity Type Id* from the drop-down for external association.
    c. Choose the *Principle Cardinality* and the *Dependent Cardinality*. Both use the following cardinality rules. Note that many-to-many relations are not supported in SAP Asset Manager
       - 0..1: Only one instance occurs; zero is also allowed
       - 1: One-to-one relations. Exactly one instance occurs
       - 0..n: Zero-to-many relations. Zero or more instances occur
       - 1..n: One-to-many relations. One or more instances occur
    d. Select the *Principle/Dependent OnDelete Cascade* checkbox, if you want to delete an associated collection when a principle or related parent entity got deleted from the mobile device. This feature only works with local objects.
    e. Type the name of your association set in the *Association Set Name* field under *Association Set*.
13. Click the *Referential Constraints* tab to add or change a referential constraint.

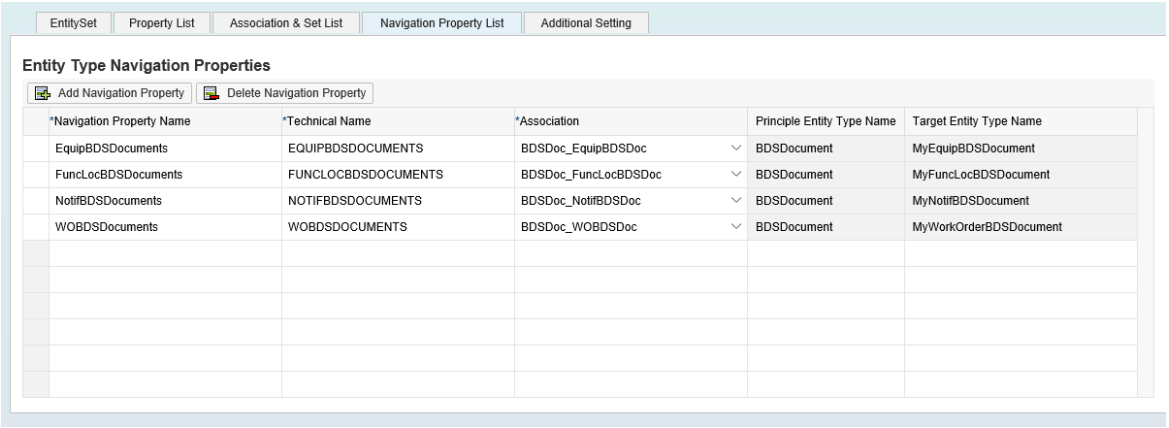


Referential constraints ensure that the principle entity that is referenced always exists. You can use a referential constraint to specify the foreign key relationship for a 1:1 relationship or a 1:n relationship.

You have to match the key properties of the principle entity type with the properties from the dependent entity type that correlates to the key property of the principle type. Populate all key properties from the principle entity type.

a. Click the *Add Constraint* button to add a new referential constraint.
b. Select the *Principle Property* and the *Dependent Property* from the drop-down menus. As mentioned earlier, the entities must be identical and all key properties must be populated from the principle entity.

14. Click the *Navigation Property List* tab to create a navigation property for entity types.



The navigation property is tied to an association, and it allows the navigation from one end of the entity type that declares the navigation property to the other related end.

> **Note**
>
> If you add a new navigation entity, first add a new association for it through the *Association & Set List*. Set the association cardinality for both principle and dependent entities.

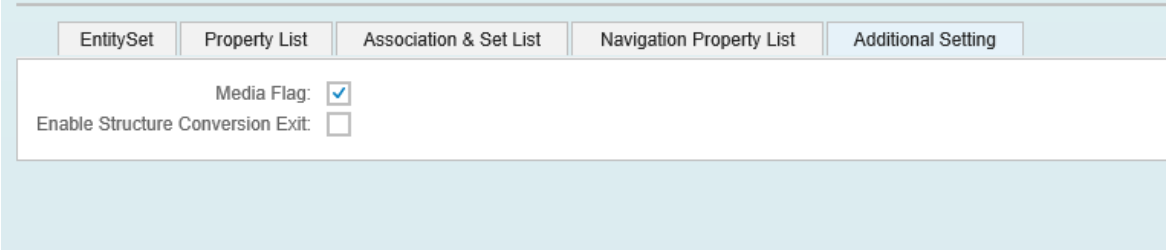
15. Click the *Add Navigation Property* to add a new navigation property.

You can create a navigation property for both principle and dependent entity type using the same association so that link will be created in both directions.

a. Type a name into the *Navigation Property Name* field.
b. Type a technical name into the *Technical Name* field.
c. Choose an *Association* (coming from the *Association & Set List* tab) from the drop-down menu.

   The *Dependent OMDO ID* and *Dependent Tech Entity Type* cells are populated based on which association entity you choose.
d. Repeat these substeps to create the navigation property on the remaining principle or dependent object.

16. Click the *Additional Setting* tab.



a. Select the *Media Flag* checkbox for media-related entity types to trigger the download of media content on the entity set collection.
b. Select the *Enable Structure Conversion Exit* checkbox to allow the SAP Asset Manager application to access the OData channel. The OData channel delegates handling of conversion exits, currency, currency amounts, units of measurement, and unit amount conversions to the SAP Gateway framework.

Results

Once the model is fully defined, when a client makes an HTTP request, it is calling for the metadata for an OData service. The SAP Gateway returns an XML string to the client, which is also reflected in the ConfigPanel.

5 Configuring SAP Asset Manager for ASPM and PdMS

5.1 Configuring Checklist Parameters

Use parameters to enable the checklist feature and configure other checklist options available.

Context

> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

The ASPM Checklist feature is called *IAM_CHECKLIST*. By default, the feature is not enabled.

To configure the checklist options for SAP Service and Asset Manager, use the *CHECKLISTS* parameter group and the following parameters within the group:

- **MobileStatusCompleted:** Default is *Completed*. Do not change this setting unless you are integrating SAP Service and Asset Manager with another product besides ASPM.
- **MobileStatusInProgress:** Default is *In Progress*
- **MobileStatusOpen:** Default is *Open*
- **CompletedStatusText:** Default is *Published*. This parameter is used to distinguish completed checklists that have been downloaded from the back end versus checklists that have been completed locally on the client but are not yet synced. The parameter is necessary to make logic decisions on the client as checklists that have been completed and synced to the back end are no longer allowed to be edited. Do not change this setting unless you are integrating SAP Service and Asset Manager with another product besides ASPM.

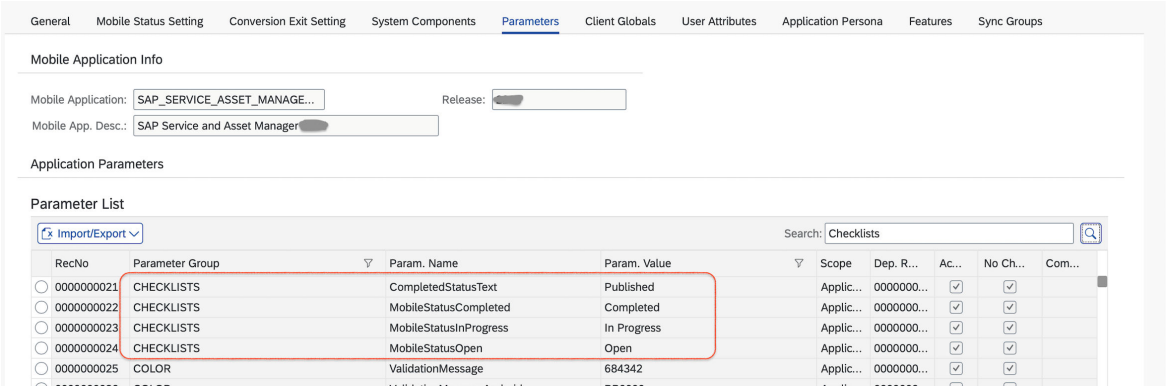
The *CHECKLISTS* parameters correspond to the rules found in the OData mobile data object *SAM2310_ASPM_CHECKLIST*. You can add a data filter rule to your customer namespace, or change the existing parameter-rule association to a new parameter-rule association.

Procedure

1. Using the ConfigPanel, navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.

2. You can scroll down to find the *CHECKLISTS* group parameters, listed in the *Context* section, or perform a search using the *Search* box. Highlight the parameter you want to configure and click the *Change* button.



3. Make your desired parameter association changes, or change the value of a parameter to *Z*, a custom activity catalog type.
4. Check the `<Active>` flag to ensure that the parameter is used by the mobile application. If desired, and if not already checked, check the `<No Runtime Change>` box to ensure that the value of the parameter is not overridden at runtime through synchronization processing.
5. *Save* your changes.

Results

You have enabled the checklist parameters in the ConfigPanel.

Next Steps

Continue to the following procedures to finish configuring the checklist feature for ASPM:

- Setting Up an ASPM Connection to the ASPM System to Use Checklists [page 253]
- Checking the Readiness of the ASPM System [page 256]
- Readiness Check for Authenticated GIS Maps [page 261]

5.2 Setting Up an ASPM Connection to the ASPM System to Use Checklists

The RFC destination is already created and connected to the ASPM out of box. However, an oAuth connection from the ASPM Cloud Foundry system to SAP Service and Asset Manager is required.

Prerequisites

Use the Integration of Asset Central Foundation with SAP EAM guide to establish an oAuth connection. Pay attention to the Server Management Properties topic.

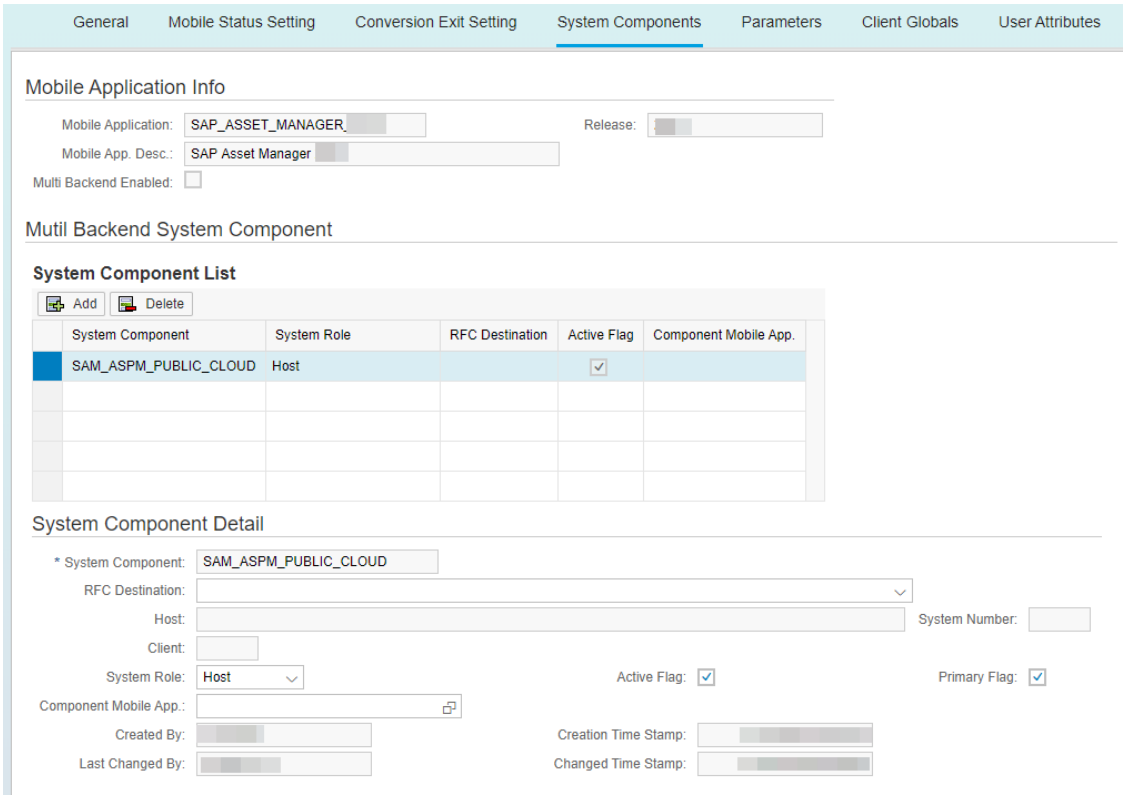
Context

Use the following procedure to create and configure an oAuth connection from the ASPM Cloud Foundry system to SAP Service and Asset Manager.

Procedure

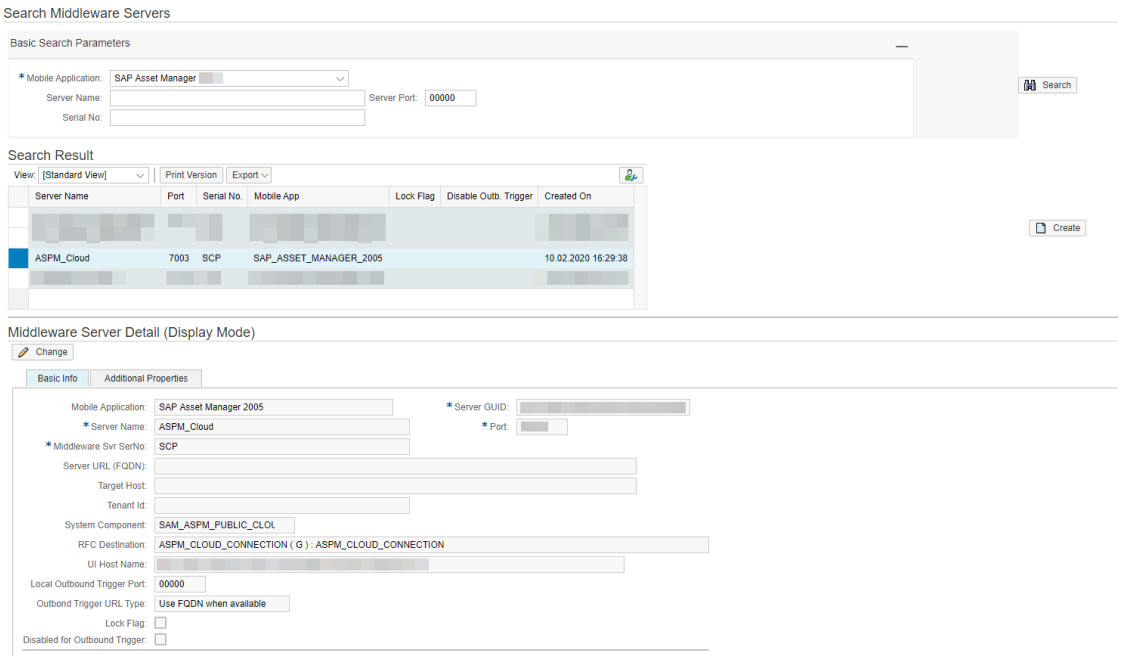
1. **Configure system component settings using the ConfigPanel:**
   a. From the home page of the ConfigPanel, select *Mobile Application Configuration*. Choose your application from the *Defined Mobile Applications* list. Then select the *System Components* tab.
   b. Click the *Change* pencil icon.

      The *System Components* page becomes editable.
   c. Click the *Add* button in the *System Component List* to add a new component.
   d. Create your new component using the following fields:
      - **System Component:** SAM_ASPM_PUBLIC_CLOUD
      - **Active Flag:** checked
      - **Primary Flag:** checked



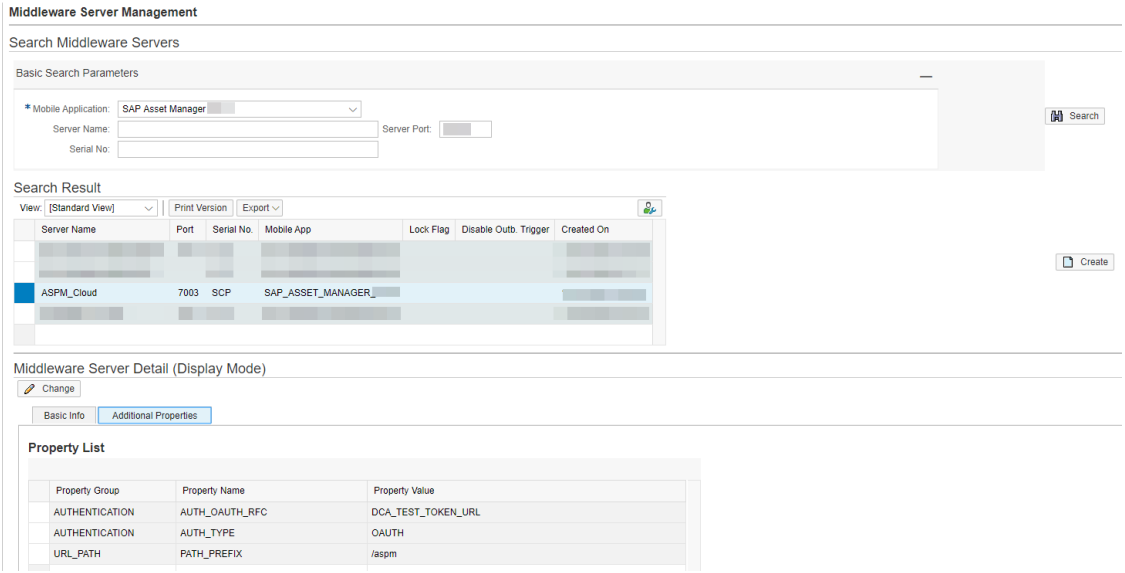
You've configured which system component you're connecting to the ASPM system.

2. Press the *Save* button. If necessary, move these changes to other systems in your landscape.
3. **Configure server component settings using the Admin Portal**
   a. Access the Admin Panel from the SAP GUI using transaction code */n/SYCLO/ADMIN*.
   b. From the main page of the Admin Portal, navigate to *Administration* > *Server Management*. In the *Search Middleware Servers* section, choose your application from the list in the `<Mobile Application>` field. Click *Search*.
   c. Click the *Create* button to add a new middleware server.
   d. In the *Middleware Server Detail* section, *Basic Info* tab, add the following fields:
      - **Server Name:** ASPM_CLOUD
      - **System Component:** SAM_ASPM_PUBIC_CLOUD
      - **RFC Destination:** [Select the ASPM RFC destination you previously created]



e. Select the *Additional Properties* tab. Add the following properties:

- Property Name: AUTH_OAUTH_RFC
  - Property Group: AUTHENTICATION
  - Property Value: [RFC destination created for oAuth token]
- Property Name: AUTH_TYPE
  - Property Group: AUTHENTICATION
  - Property Value: OAUTH
- Property Name: PATH_PREFIX
  - Property Group: URL_PATH
  - Property Value: [/ain or /aspm depended on Asset Central API version]



You've configured which ASPM system you're connecting to.

4. Click the *Save* button to save your new settings.

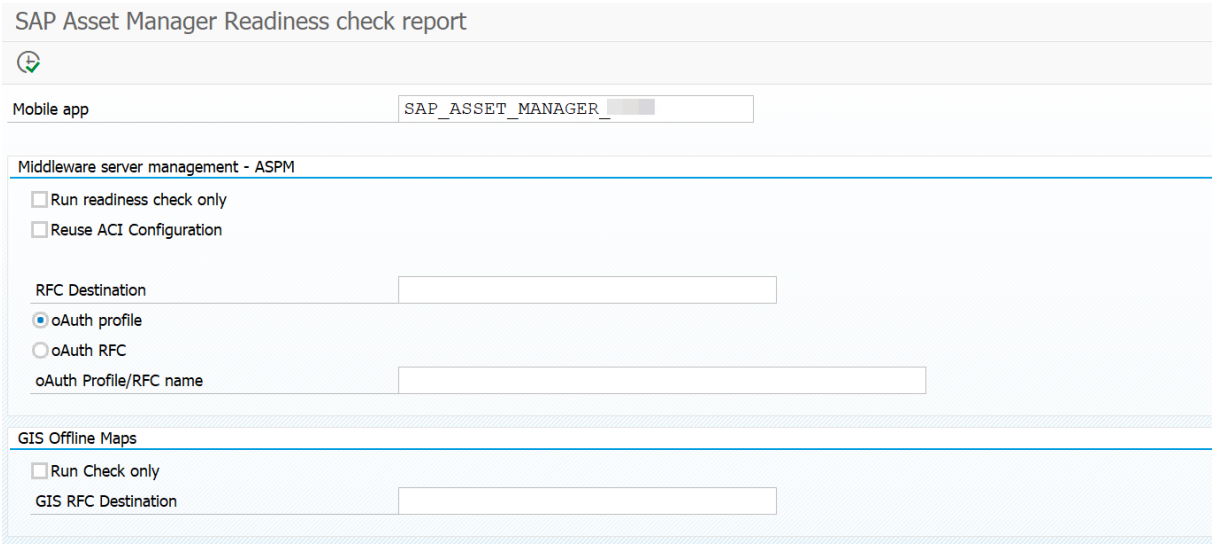
5.2.1 Checking the Readiness of the ASPM System

The ASPM readiness check program on the SAP GUI checks the readiness of the ASPM checklist feature.

> **Note**
>
> Before following the instructions in this topic, ensure that you've performed the Setting Up an ASPM Connection to the ASPM System to Use Checklists [page 253] procedure.

The readiness check program checks your set-up (performed in the Setting Up an ASPM Connection to the ASPM System to Use Checklists [page 253] procedure), such as the system component. If your configuration has an error, a red light displays on the output that isn't set up properly. The /syclo/admin set-up is also completed you've selected certain options listed in this topic.

Using the SAP GUI, execute the program */MERP/CORE_READINESS_CHK_PROG* to run the readiness check. The selections you make in this procedure are all found in the main *Middleware server management - ASPM* section. See the Readiness Check for Authenticated GIS Maps [page 261] procedure for details on the *GIS Offline Maps* section.

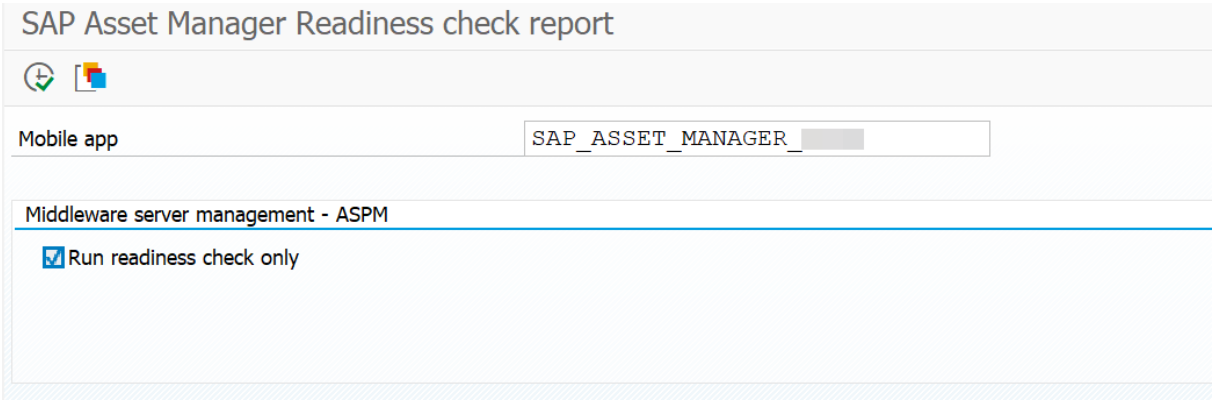


The readiness check program gives 3 options to check or complete the ASPM checklist for SAP Service and Asset Manager:

Run Readiness Check Only

This checkbox disables the rest of the inputs on the screen if checked. Choosing this option runs the program in check-only mode. No RFC destination or configuration is made. Select this option if you've performed all the steps in Setting Up an ASPM Connection to the ASPM System to Use Checklists [page 253]. Running the readiness check only confirms that no steps were missed. If there are errors in your set-up, a red light displays by the output. You can then use the transaction code for that output to fix the issue.

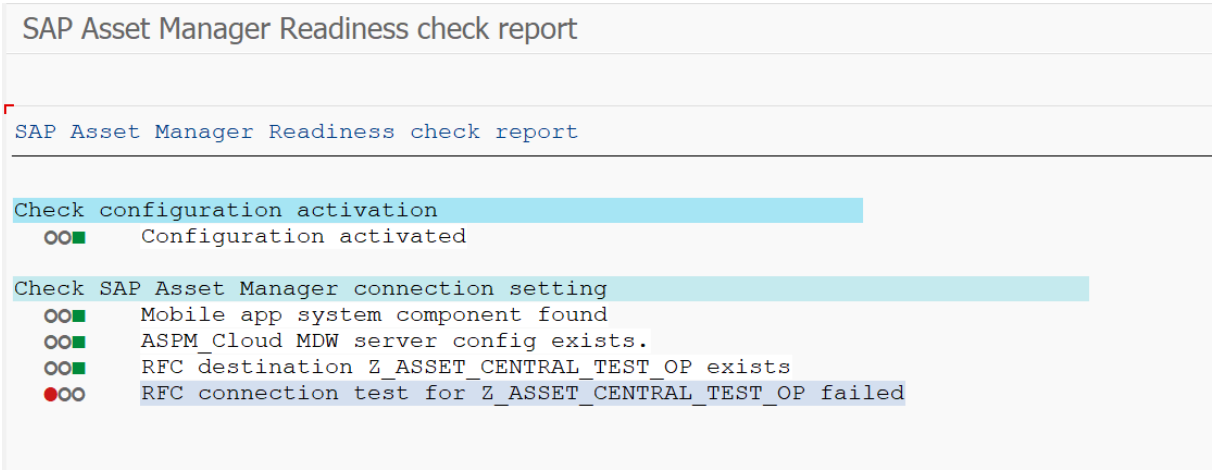
Input



Output

In the following sample screenshot, the red light result means that the RFC connection test used to set up the ASPM checklist feature has failed the connection test. To fix:

1. Enter transaction code *SM59* and test the connection.
2. If the connection fails, fix the issue by using a proxy. Make sure that configurations such as user credentials are correct.
3. Rerun the check program after fixing any issues to ensure a green light is displayed.

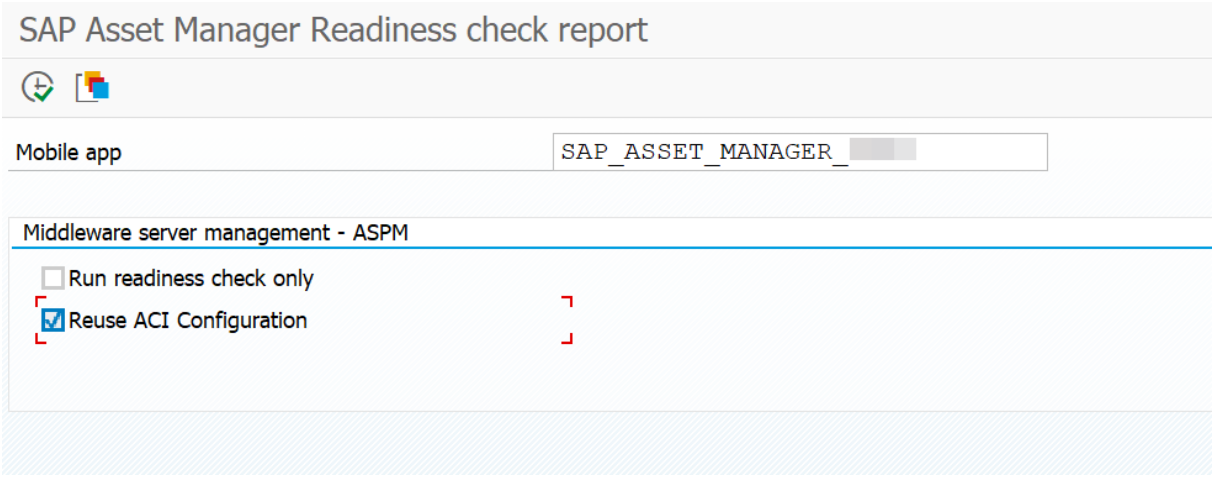


Reuse Asset Central Integration Configuration

Use this checkbox to reuse the ACI app if it exists. Selecting this option uses the same RFC destination and authentication properties to create a new server configuration.

Input

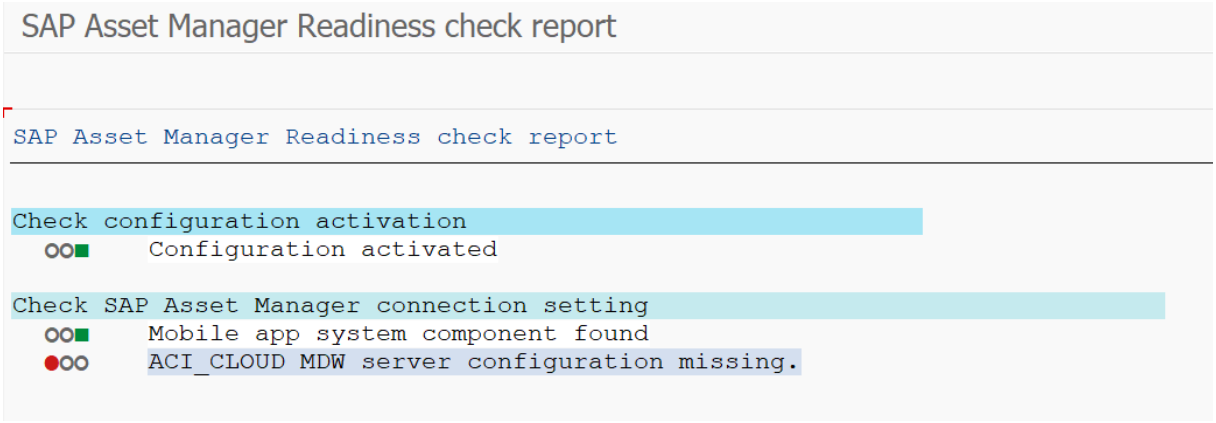
Use this option if *Step 3d* wasn't completed in the Setting Up an ASPM Connection to the ASPM System to Use Checklists [page 253] procedure and you want to result the RFC destination used for ACI configuration to complete the set-up .



Error Output Example

This option uses the ACI middleware configuration to create the ASPM_CLOUD server configuration for SAP Service and Asset Manager. If ACI configuration doesn't exist, use the *Input RFC Destination* option to create the configuration.

The red light shown in the following example screenshot shows the error thrown when there's no ACI configuration in the customer system to reuse. If a red light is displayed, complete the set-up, then run the check program in reuse mode to complete the ASPM checklist set.



Input RFC Destination

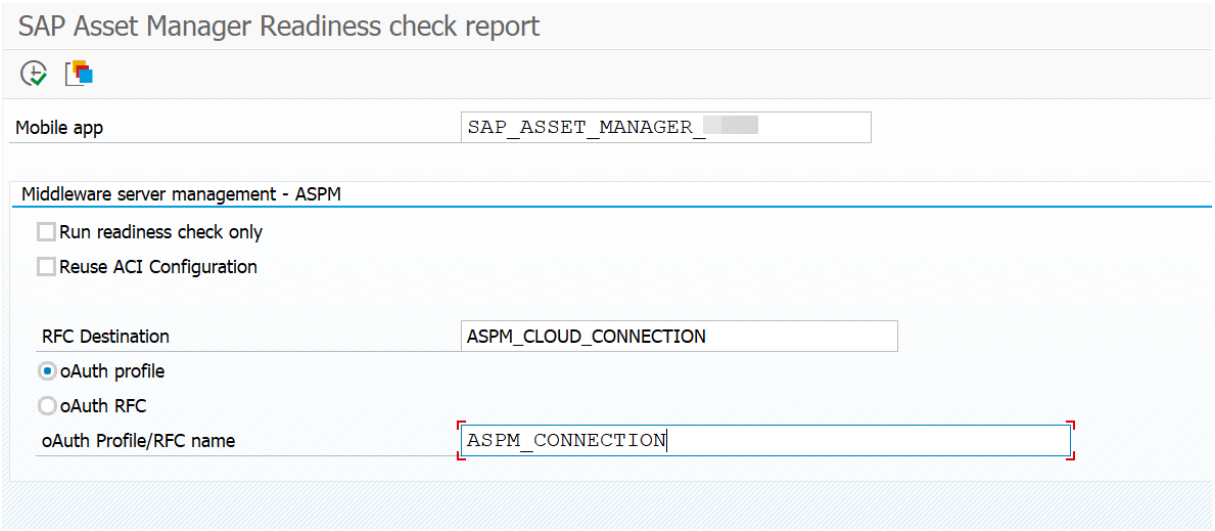
Create the RFC destination or middleware server configuration: Using the *RFC Destination* section, give an *RFC Destination*, select an *oAuth Method* and *oAuth Value*. An RFC destination and/or middleware server configuration is created if they don't already exist.

Use this option if *Step 3d* wasn't completed in the Setting Up an ASPM Connection to the ASPM System to Use Checklists [page 253] procedure and you want to complete the rest of the set-up-up, along with the RFC destination creation. Once the destination is created, or if the destination exists, but the ASPM feature set up in */SYCLO/ADMIN* doesn't exist, this program completes the set-up with the RFC destination.

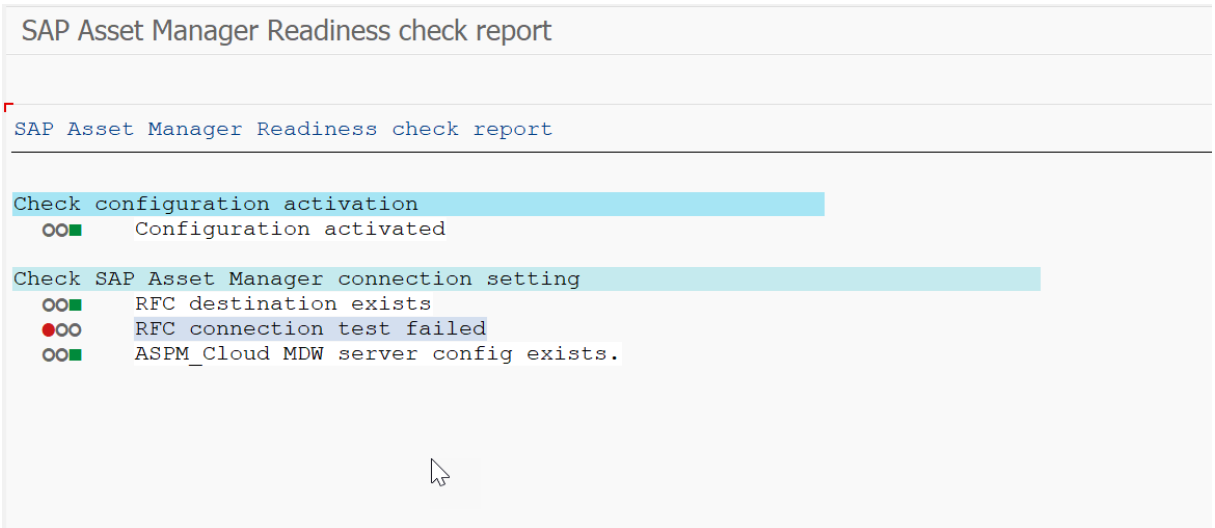
If the RFC destination and middleware configuration don't exist; the program redirects to the *RFC Destination Creation* screen. Once the RFC destination is ready, it creates a management entry for the middleware server in the configuration tables for SAP Service and Asset Manager.

> **Note**
>
> For SAP_BASIS release 752 and above, use the radio button *oAuth Profile* and property *AUTH_OAUTH_PROFILE*. For earlier versions, use the radio button *oAuth RFC* and property *AUTH_OAUTH_RFC*.

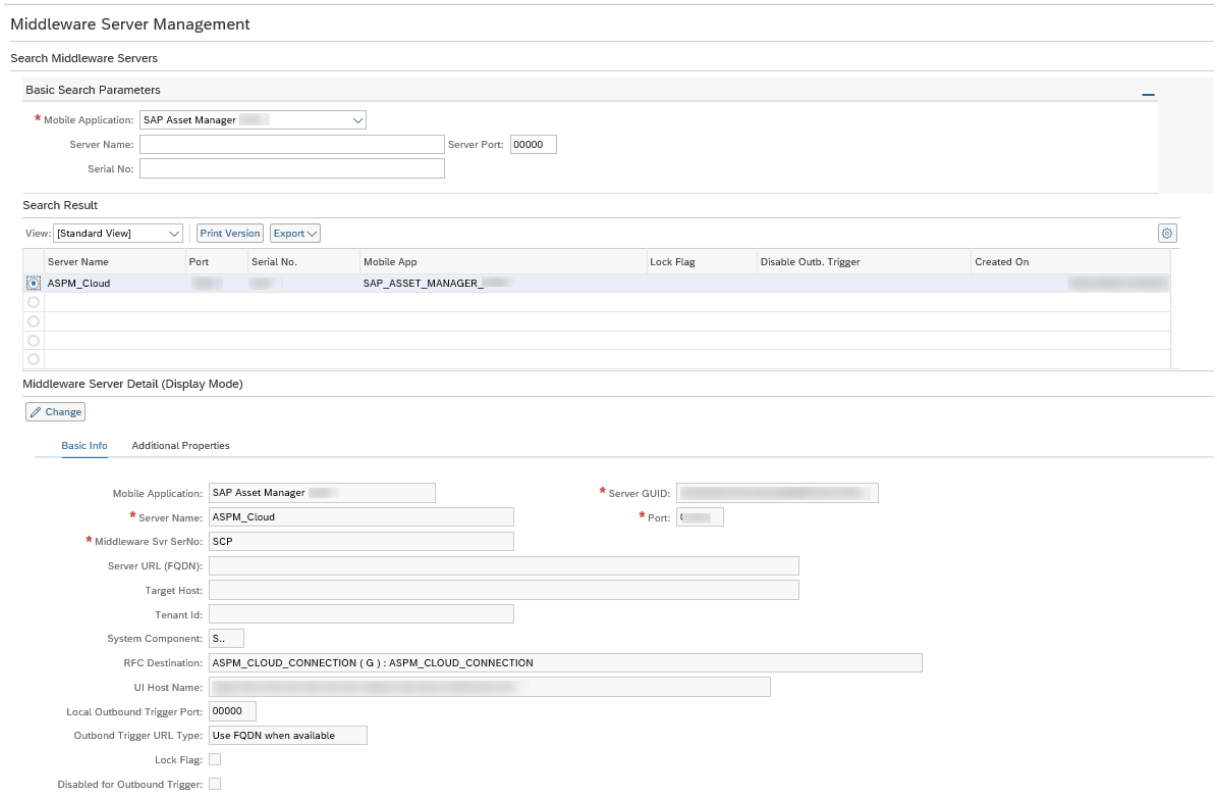
Input



Output



The */SYCLO/ADMIN* ConfigPanel entry the ASPM readiness check program checks or creates is shown in the following example:



Troubleshooting

If the output of the readiness check shows the RFC connection test failed, perform a manual test. Make sure to add a proxy and/or certificate to *STRUST* if needed.

You can manually change the middleware server configuration via */SYCLO/ADMIN*.

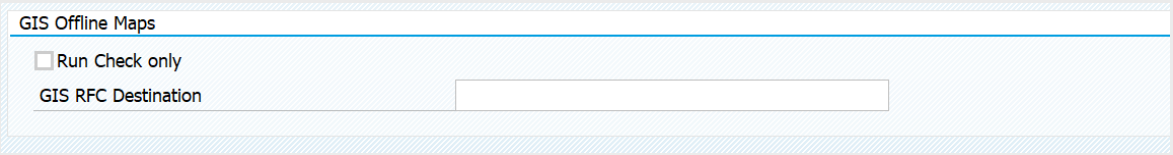
Next Steps

If you're using SAP Service and Asset Manager with authenticated GIS, continue to the Readiness Check for Authenticated GIS Maps [page 261] topic.

5.2.1.1 Readiness Check for Authenticated GIS Maps

Context

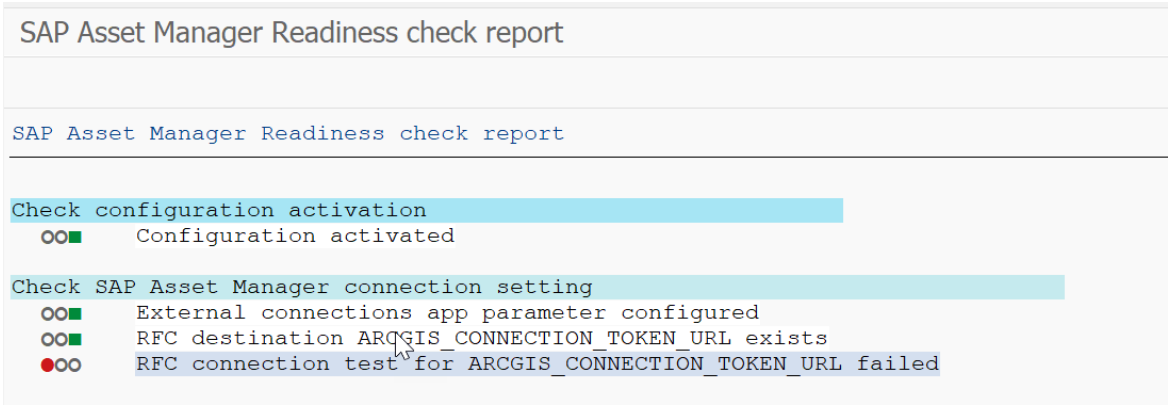
> **Note**
>
> Before performing this procedure, follow the appropriate ASPM readiness check. See the Checking the Readiness of the ASPM System [page 256] topic for details.



Procedure

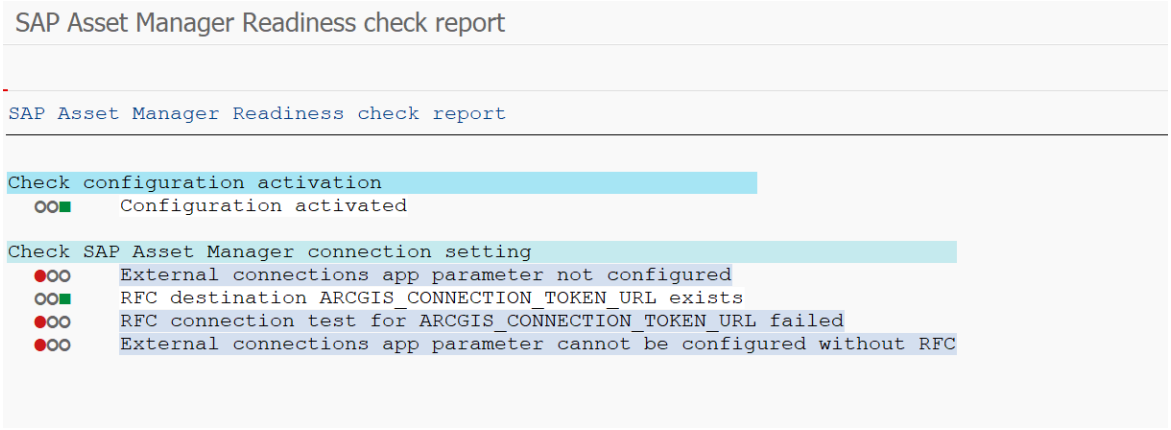
1. **Run Readiness Check Only**

   Example of *Run Check only* output:

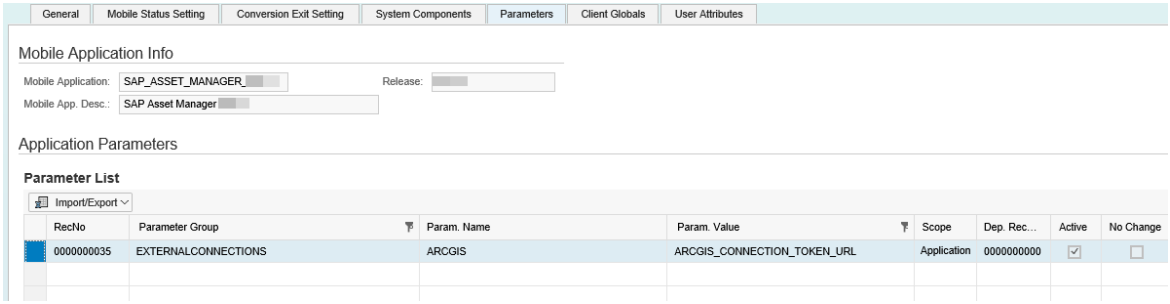


2. **Create the RFC Destination:** Creating the RFC destination stores the GIS server login credentials and adds them to the mobile application parameter.

Example of *GIS RFC Destination* and config parameter output:



The */SYCLO/ADMIN* ConfigPanel entry the Authenticated GIS readiness check program checks or creates is shown in the following example:



6 Field Operations Worker Configuration

SAP Service and Asset Manager for Field Operations Worker uses the digital core with SAP S/4HANA for task driven activities and rounds. It supports workers who perform asset inspections and checks with focus on measurement points and on smaller services and repairs.

Field Operations Worker, or FOW, is an add-on component to SAP Service and Asset Manager. If you don't see FOW features while using the SAP Service and Asset Manager application, or in the ConfigPanel, your site hasn't installed the component.

Field Operations Worker adds the following functionality to the core SAP Service and Asset Manager application:

- **View routes data:** A route is comparable to a work order in the base SAP Service and Asset Manager application.
- **View stops data:** A stop in Field Operations Worker is comparable to an operation in the base SAP Service and Asset Manager application. A route is composed of one or more stops.
- **View asset information:** An asset in Field Operations Worker is comparable to a piece of equipment in the base SAP Service and Asset Manager application. Assets are located at an FOW stop.
- Use field data capture to take readings on measurement points. Measurement points are located on assets or a set of assets at a route stop.

6.1 Differentiating a Base SAP Service and Asset Manager Work Order from a Field Operations Worker Order

Creating rules based on order types affects synchronization processing and order downloads to the mobile devices of your users who use the Field Operations Worker component.

Prerequisites

Address the following before performing the procedure:

- The order types for work orders that are downloaded to technicians using the Field Operations Worker component are already determined.
- The person performing the procedure has access to the Config Panel and permissions to change settings.

Context

> **Note**
>
> Field Operations Worker orders are a subset of the base SAP Service and Asset Manager application work orders.

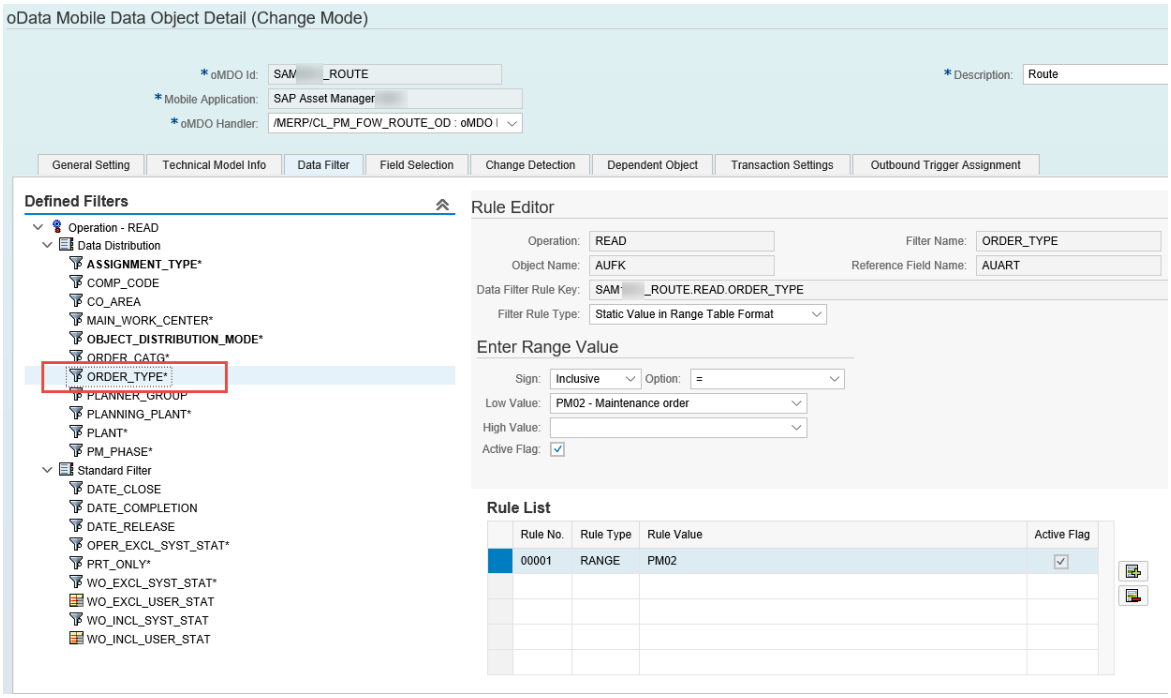
The following procedure modifies the synchronizing behavior of the SAP Service and Asset Manager application, along with the Field Operations Worker component. After you complete the procedure, only orders with a given order type of *PM02* are downloaded to the FOW component. In the procedure, you change the *ORDER_TYPE* filter in the OMDOs involved in order synchronization. Specifically, you add a rule to the filter in the **SAM2310_ROUTE** OMDO to include only the desired order type.

If you don't create a rule for the *PM02* order type, then that order type is excluded from work order download synchronization processing. If the FOW orders are excluded from synchronization processing, then the orders aren't present on the mobile clients of your users.

Procedure

1. Click the *OData Mobile Data Object Configuration* link, then click the *Data Filter* tab from the main ConfigPanel page. Be sure to have your desired mobile application chosen in the *Mobile Application Filter* field at the top of the page.

   Selecting an application filters the *OData Mobile Data Object by Mobile App* choices in the left panel with only OMDOs available in your application.
2. Expand the *OData Mobile Data Object by Mobile App* list on the left and click **SAM2310_ROUTE**.
3. Select the *Data Filter* tab.
4. Click the *Operation - READ* > *Data Distribution* > *ORDER_TYPE* node in the *Defined Filters* list.
5. Create a rule using the following parameters if the rule doesn't already exist:
   - **DOF Rule Type:** Static Value in Range Format
   - **Sign:** Inclusive
   - **Option:** =
   - **Low Value:** PM02 - Maintenance order
   - Ensure the *Active Flag* box is checked



6. Influence Field Operations Worker orders using the following filters found in *Operation - READ* > *Standard Filter* if needed:
   - WO_EXCL_SYST_STAT
   - WO_EXCL_USER_STAT
   - WO_INCL_SYST_STAT
   - WO_INCL_USER_STAT
7. *Save* your changes.

Results

After you finish the procedure, both Field Operations Worker orders and base SAP Service and Asset Manager work orders are downloaded by the SAP Service and Asset Manager application.

7 Safety Technician Configuration

7.1 Configuration Prerequisites and Limitations - Safety Technician Persona

Prerequisites

There are a few guidelines that must be applied to the SAP system for the user to be able to use the safety technician persona:

- The https://me.sap.com/notes/3262754 SAP note: On ECC, SMERP software component, Safety Technician persona is available **only on SP13 or higher**.
- The https://launchpad.support.sap.com/#/notes/2551483 SAP note: No transfer of lock number for item maintenance for API.
  Lock numbers are not transmitted during the maintenance of existing items in the work clearance document using the API used by the SAP Service and Asset Manager mobile application. This is necessary because safety technician persona can enter the physical lock number during the tagging process.
- The https://launchpad.support.sap.com/#/notes/3277604 SAP note: Data provision for Asset Management.
  Use BAdI/Exit to effectively delta-synchronize the WCM data for SAP Service and Asset Manager.
- The https://launchpad.support.sap.com/#/notes/2431765 SAP note: Processing the user status in the APIs for WCM objects.
  Use the WCM APIs to execute function codes (ACTION_EXECUTE suffix) to support setting or resetting the user status. User status can be entered in Mobile Status configuration at the Syclo Configpanel.

Limitations

Currently, the Safety Technician persona has the following limitations:

- When creating and configuring usage types [page 270]:
  - The following Operational Cycle types are supported:
    - Tagging cycle without temporary untagging phase
    - No items
  - The following Operational Cycle types are **not** supported:
    - Tagging cycle with temporary untagging phase
    - Test cycle
- When creating and configuring work orders as dependent objects [page 271], assignments at the operation level are **not** supported, only at work order header level.

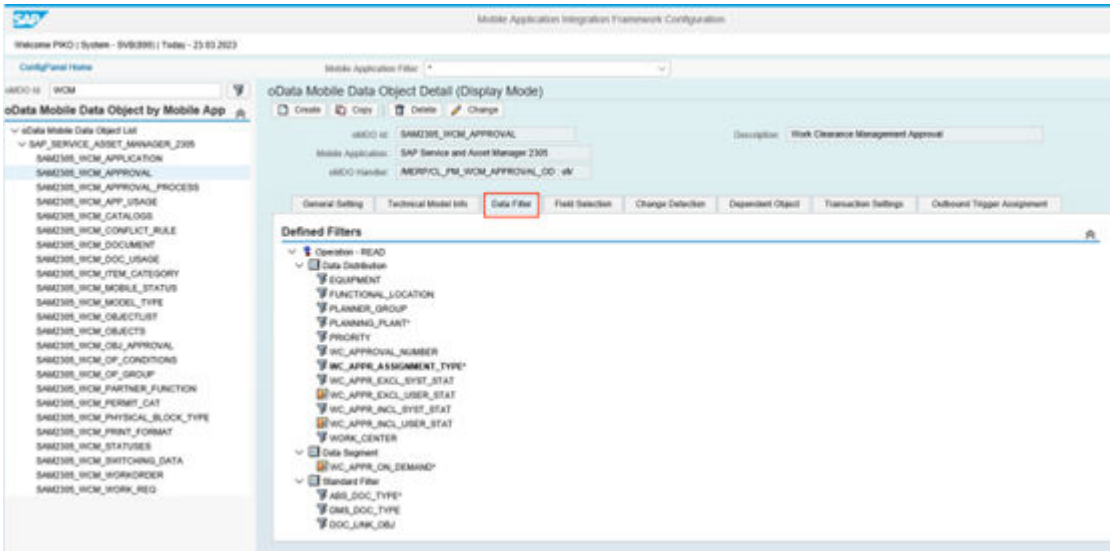
7.2 oMDO Configuration

Context

To change the oMDO data filters, follow these steps:

Procedure

1. In the SAP GUI, navigate to the `/n/syclo/configpanel` transaction to open the syclo configuration panel.
2. Under *oData Channel Integration Settings*, navigate to *oData Mobile Data Configuration*.
3. By selecting the *oMdo* required, you will find the filters in the *Data Filter* section.
4. Navigate to *Change* mode and add any data filter.



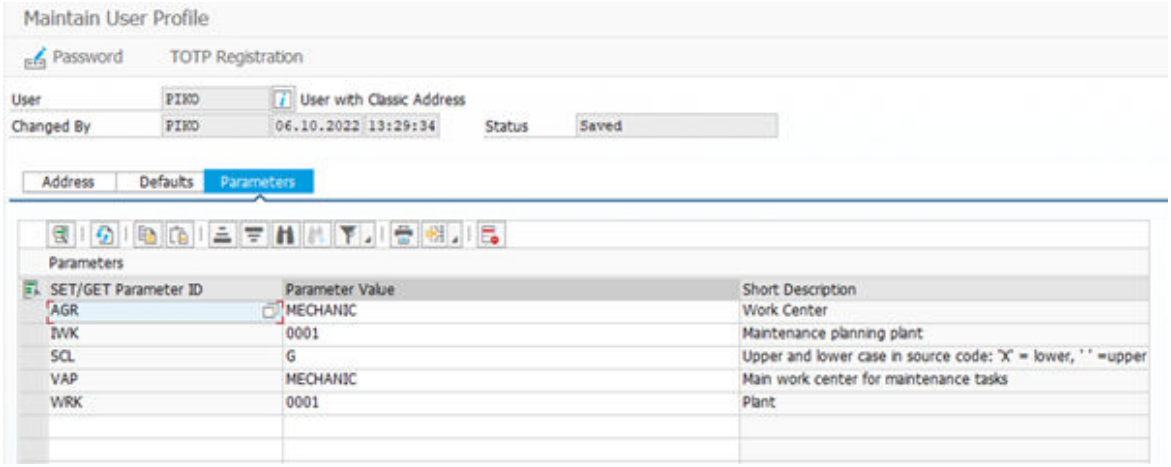
Assignment types

The following work permit assignment types (same for safety certificates) are supported:

- 1: User Plant (the user must add the IWK user parameter)
- 2: User Work Center (the user must add the VAP user parameter)
- 3: User Planner Group (the user must have the IHG user parameter setup)
- 4: Partner Function
- 5: Selection Variant
- 6: Operational List + Selection Variant
- D: Dependency queue

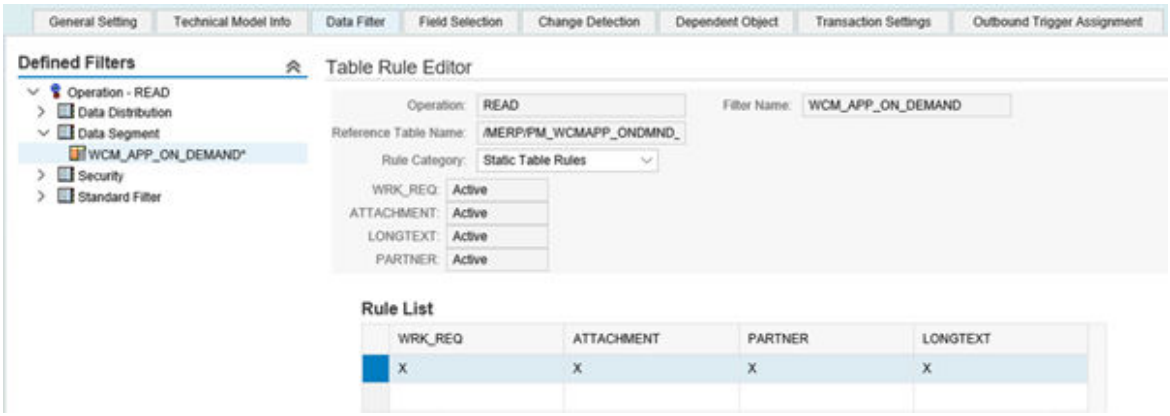
- Z: Other (implement BADI)

5. On the SAP GUI, navigate to the *SU3* transaction and select the *Parameters* tab.
6. Add a user parameter identifier with the value.



On demand structure configuration

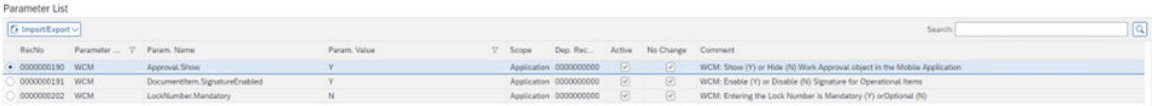
It is possible to disable or enable sampling of some entities if required. For example, you can disable `WCMApplicationLongtext`, `WCMApplicationAttachments` and other such entities.



7.3 Mobile Application Parameters

Use specific mobile app settings to get some entities disappear and make the signature optional. You can change these settings on the *Mobile App Configuration* page.

1. Navigate to the *Syclo ConfigPanel* and then select *Mobile Application Configuration*
2. Navigate to *Mobile App* and then select the *Parameters* tab.
   The possible values for safety technician persona are as follows (see figure below):



There are three available options for using the application:

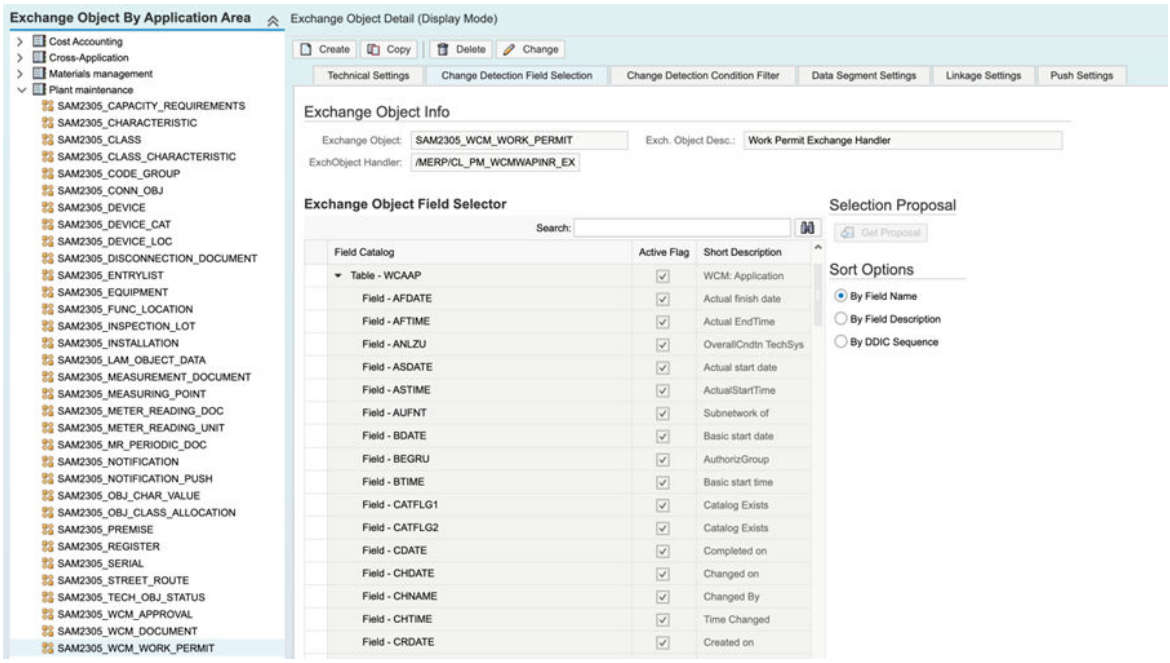
- `Approval.Show`. Enables or disables the display of Work Approval objects
- `DocumentItem.SignatureEnabled`. Enables or disables signature input during tagging or untagging of an operational item. When disabled, it also hides the signature panel.
- `LockNumber.Mandatory`. Users can set the Lock Number as mandatory or optional in the tagging process.

7.4 xChange

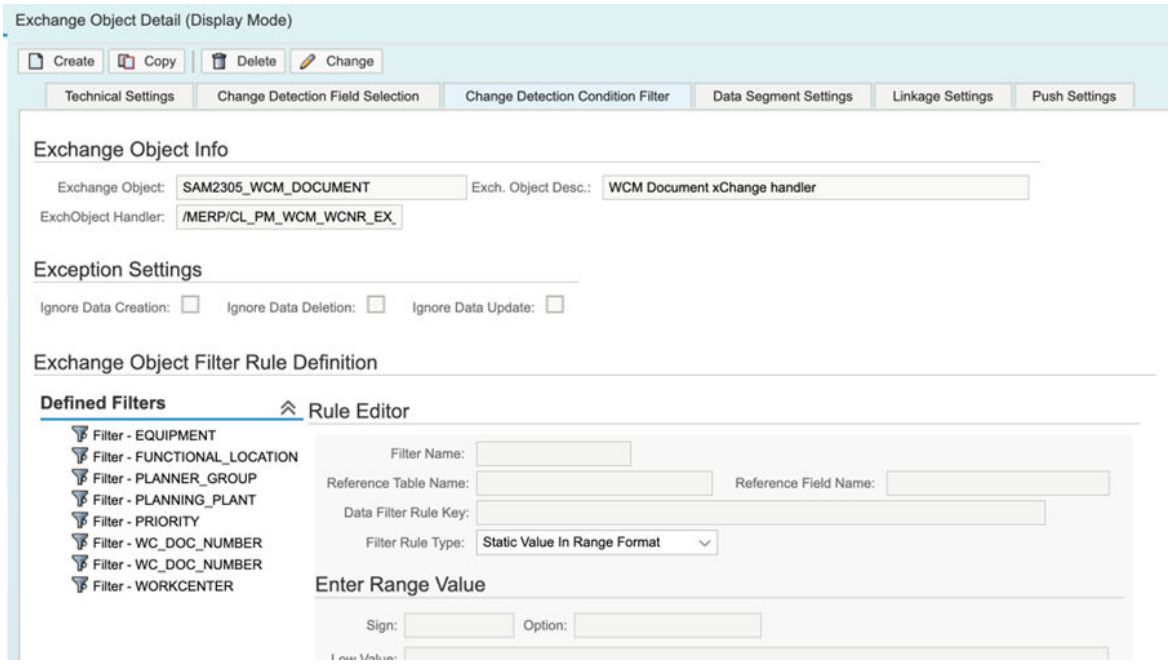
With the exchange framework, we also provide exchange detection. The exchange framework allows users to capture changes made in the backend to WCM objects. This is also highly customizable. You can choose which fields you want to check, and you can also enter data filters.

To access the xChange field detection and select specific fields of the object:

1. Navigate to the *Configuration panel*, select *Exchange Object Configuration*.
2. Navigate to *Plant Maintenance* and select the *SAM2305*WCM** objects.



You can also add data filters similar to oMdos:

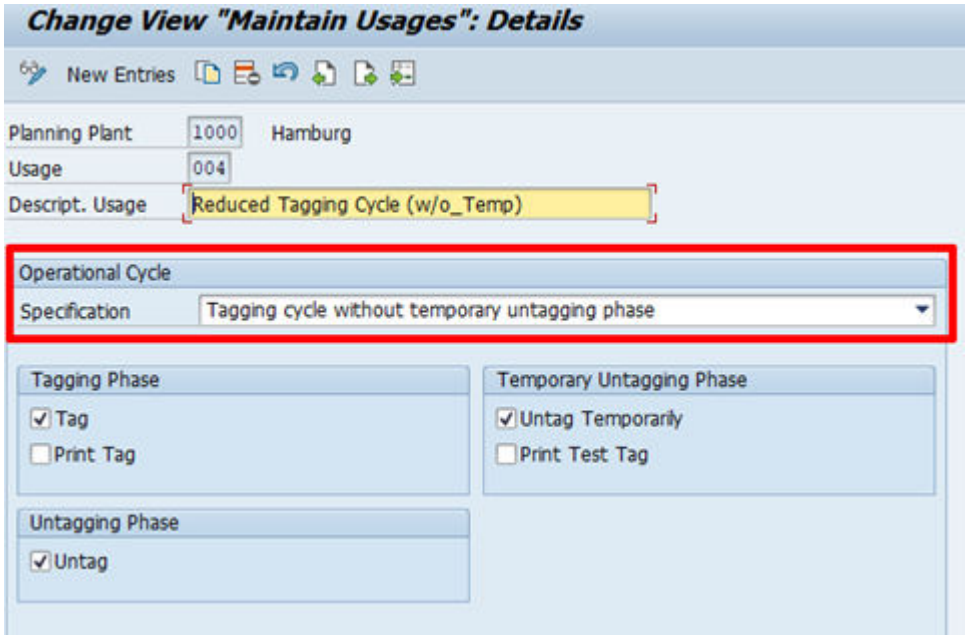


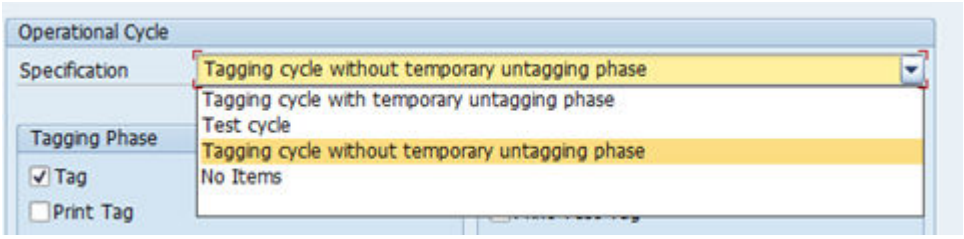
7.5 Supported usage types

You can create and configure the usage types. There are several customization options, the most important of which is *Operation Cycle - Specification*.

The current release supports the following two types:

- Tagging cycle without temporary untagging phase
- No items.





7.6 Work Order as a Dependency

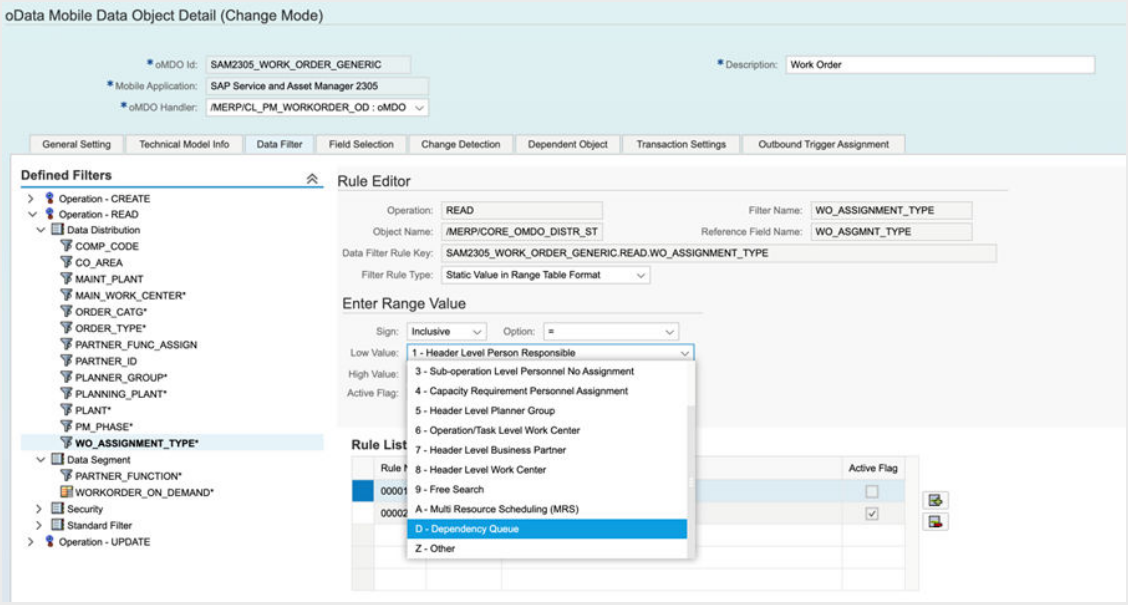
A new release related to the safety technician persona is that work orders can be dependent objects. This means that if the persona is enabled, you have the ability to load work orders that are associated with the related work permits.

Note

In this release, only assignments at the work order header level are supported for the safety technician persona. Operation level assignments will be supported in a later version.

To enable this configuration:

1. Navigate to the `Configuration panel`, select *oMDO* configuration, then select `SAM****_WORK_ORDER_GENERIC`
2. Navigate to *Data Filters* and add a *D Assignment Type* to the object.



7.7 Assignment Management

Safety Technicians can see a filtered list of items on their mobile devices that are assigned to them (see Filtering Your Assigned Objects). This applies to the following list screens:

- *Operational Items*
- *Isolation Certificates*
- *Work Permits*
- *Work Orders*
- *Work Approvals*

You can enable the *Assigned to me* filtering option for users by the following Assignment Types:

- *4 - Partner Function (Approval Roles + Partner Functions)*: See Assignment Configuration by Assignment Type 4 - Partner Function [page 272]
- *5 - Selection Variant*: See Assignment Configuration by Assignment Type 5 - Selection Variant [page 278]
- *6 - Operational List (Operational list + Selection Variant)*: See Assignment Configuration by Assignment Type 6 - Operational List + Selection Variant [page 283]

> ⓘ Note
>
> *5 - Selection Variant* and *6 - Operational List (Operational list + Selection Variant)* are non-regular Assignment Types, dedicated to provide additional assignment functionalities. You can create selection variants, define filters for them, and assign the variant to users. Then the filters from the variant will be used during a sync. When only these assignment types are set up, regular oMDO filters are not taken into account during the sync.

7.7.1 Assignment Configuration by Assignment Type 4 - Partner Function

Context

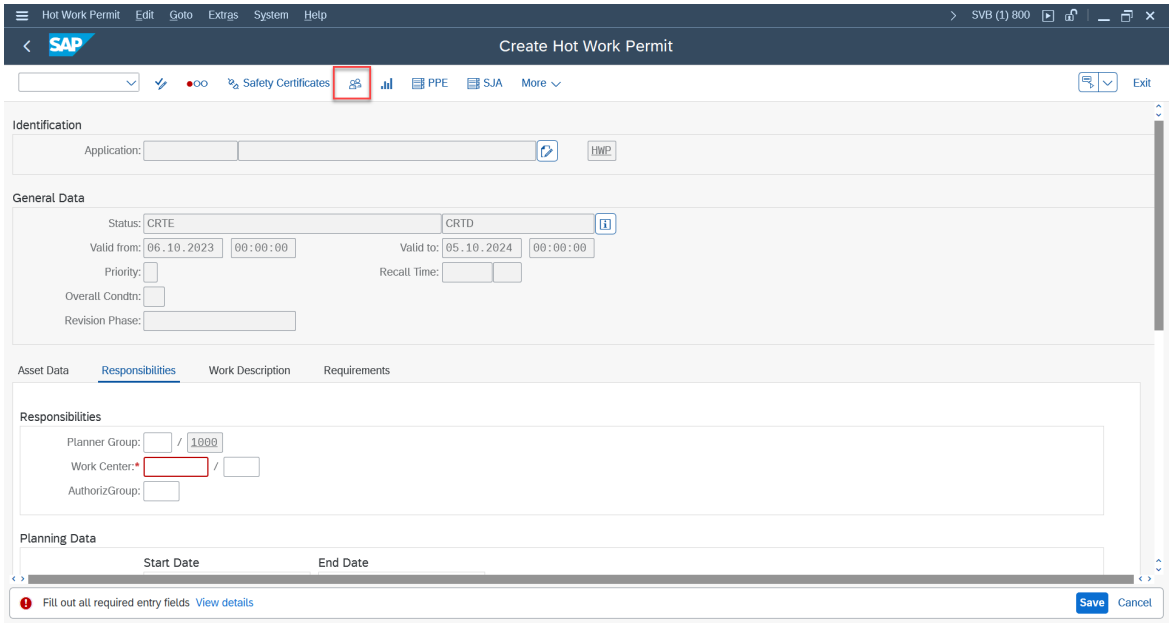
To enable the *Assigned to me* filtering option for users by the *4 - Partner Function (Approval Roles + Partner Functions* Assignment Type, perform the following steps:

> ⓘ Note
>
> - This procedure applies to Work Permits (and related Work Orders, Work approvals) and Isolation Certificates (and related Operational Items).
> - Regarding the order of steps, it does not matter if you perform the SAP GUI or Configuration Panel changes first.
> - If you want to enable for Safety Technicians to filter for objects assigned to other users as well, perform the additional steps from Configuring the ShowAssignedToList parameter [page 276].

Procedure

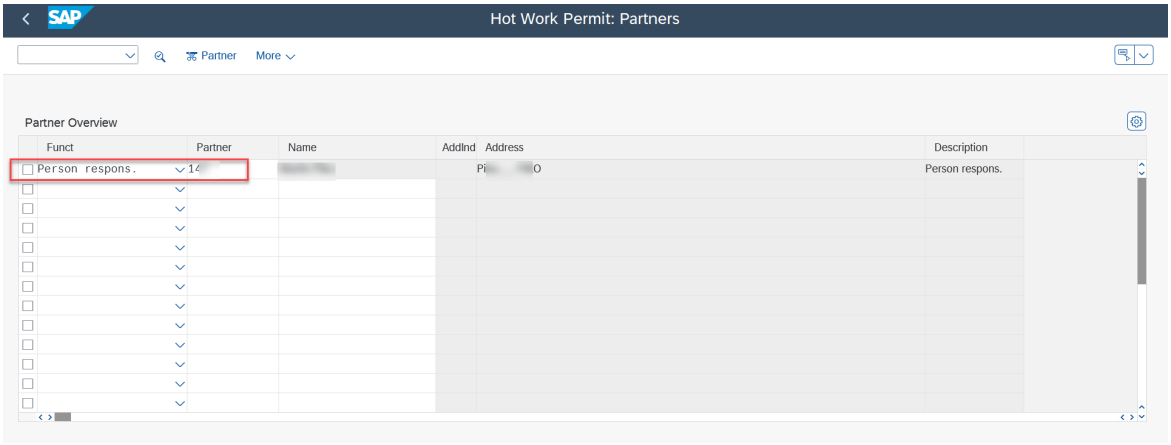
SAP GUI changes

1. Create a new (or change existing) Work Permit or Safety Cerificate.
   - To create a new **Work Permit**, use the transaction code `WCTK`.
   - To create a new **Safety Cerificate**, use the transaction code `WCT6`.
2. After the Initial Screen, click the *Create Partner* icon at the top of the Create Work Permit / Safety Cerificate screen.



> ### Note
>
> To do that, first you have to fill in the *Work Center* field.

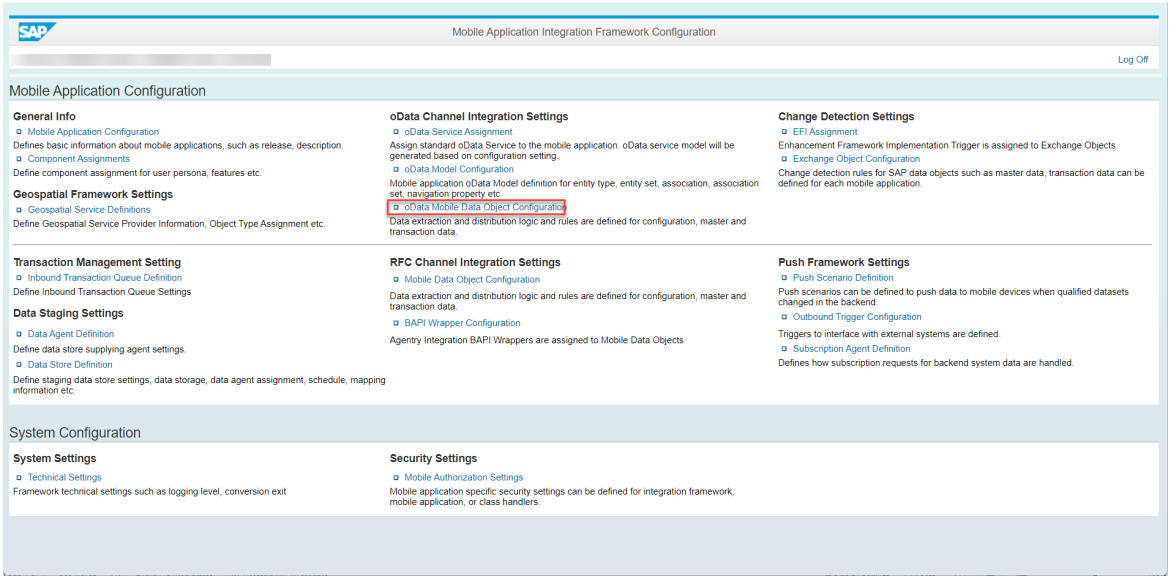
3. On the *Partners* screen, fill in partner data as follows:
   1. *Partner Function*: From the drop-down, select *Person Responsible*.
   2. *Partner*: Enter the Personal Number of the user, who you would like to assign to the Work Permit / Safety Cerificate.
   3. All other fields will be filled automatically.



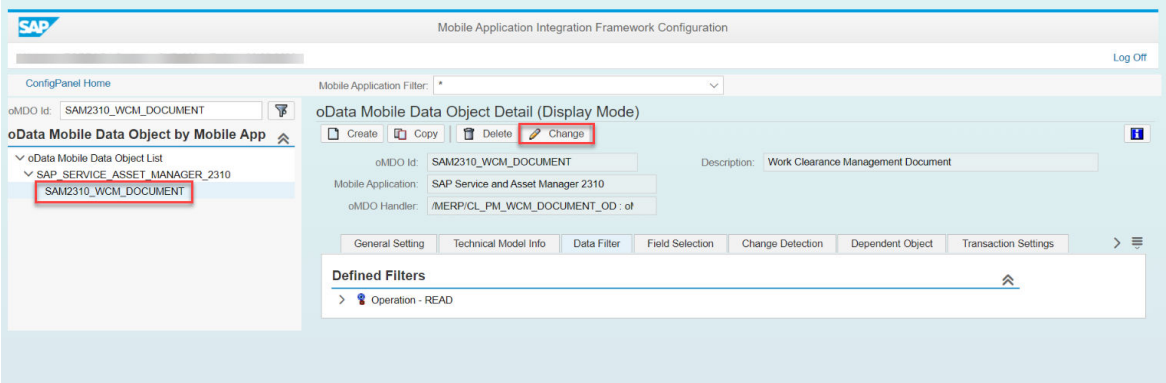
> **Note**
>
> - If the user does not have a Personal Number yet, you need to create it in the SAP HR Module.
> - By using *Person Responsible* as *Partner Function*, you can assign a specific SAP user to the Work Permit / Safety Cerificate.

Configuration Panel changes

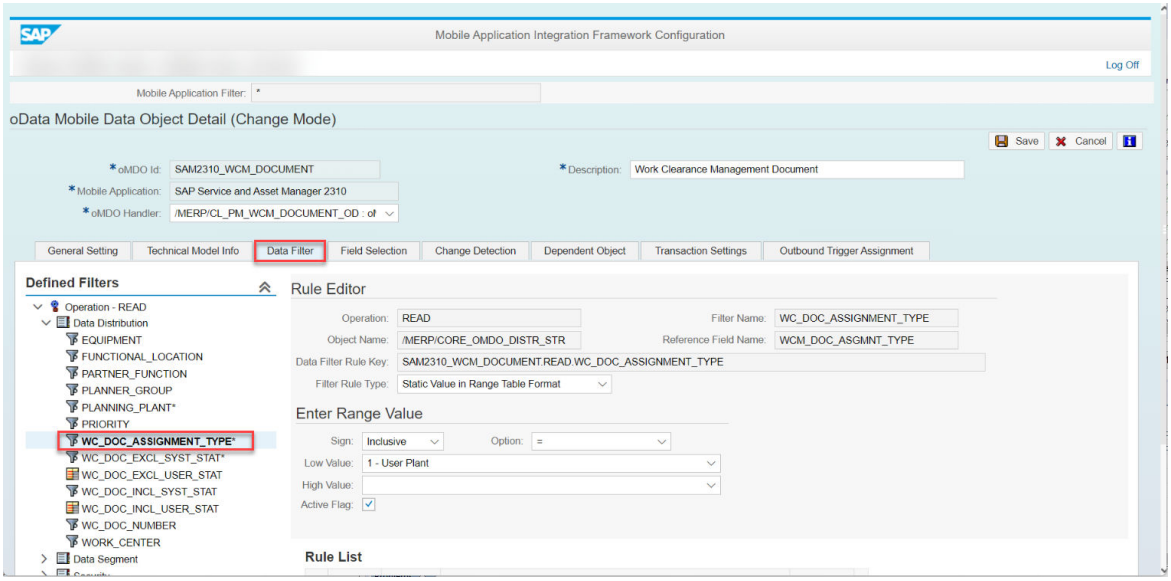
4. In the *SAP GUI*, navigate to the `/n/syclo/configpanel` transaction to open the Configuration Panel.
5. Under *oData Channel Integration Settings*, navigate to *oData Mobile Data Configuration*.



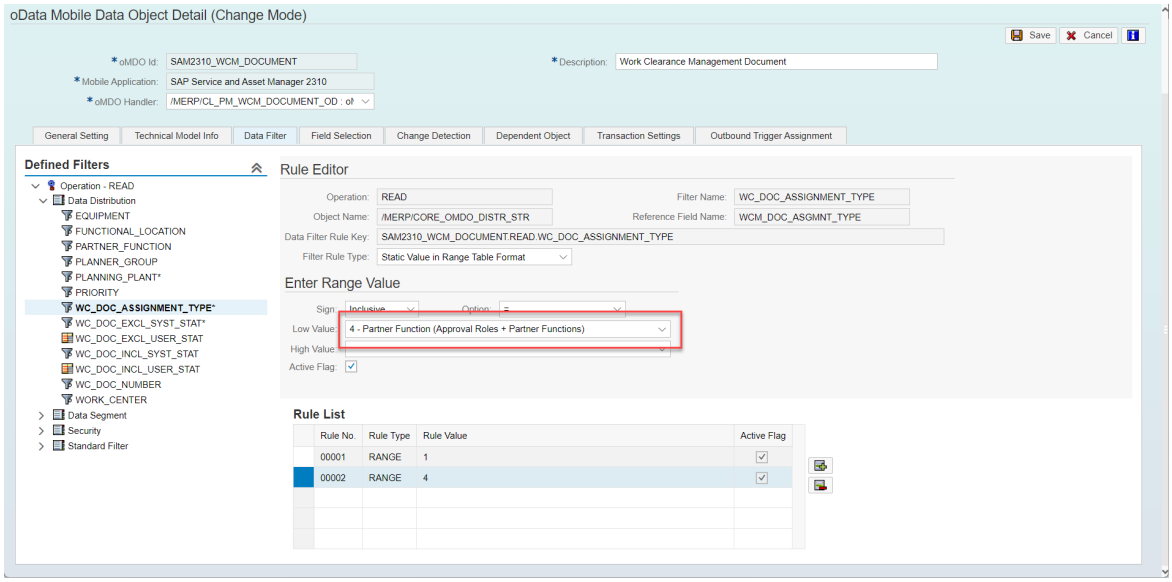
6. From the list of oMDOs, select the relevant object and click *Change*:
   - *SAM2310_WCM_DOCUMENT* for Safety Cerificates
   - *SAM2310_WCM_APPLICATION* for Work Permits



7. Under the *Data Filter* tab, select:
   - *WC_DOC_ASSIGNMENT_TYPE* for Safety Cerificates
   - *WC_APP_ASSIGNMENT_TYPE* for Work Permits

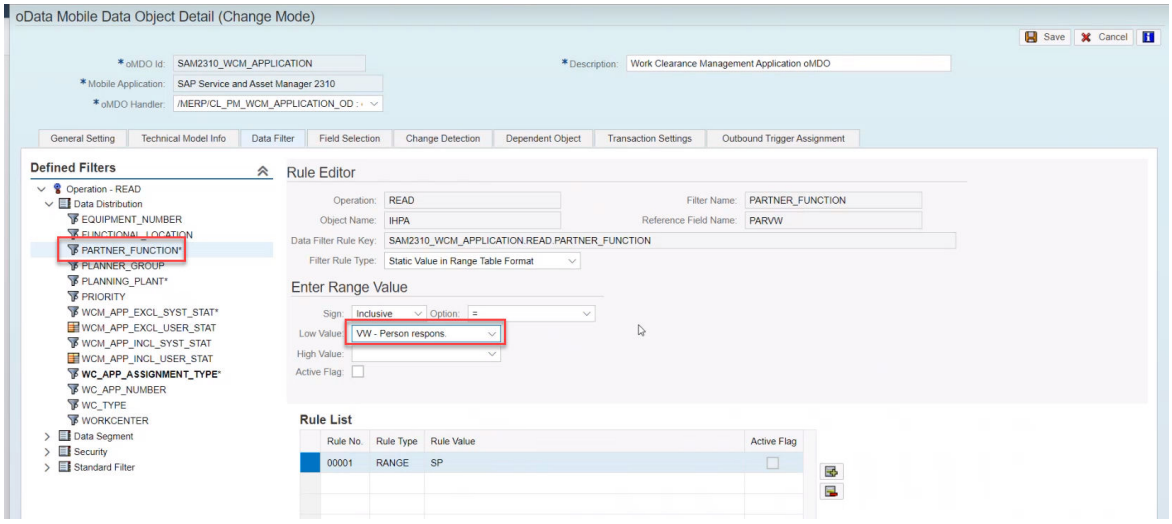


8. From the Low Value drop-down list, choose *4 - Partner Function (Approval Roles + Partner Functions)* as assignment type.



> ⓘ Note
>
> Consider the following in the *Rule List* table, *Rule Value* column:
>
> - If there is only one row with assignment type *4*, then only the objects assigned to the user will be downloaded to the client (*Assigned to me* button will not be displayed).
> - If there are two rows with assignment types *1* and *4* (see the screenshot above), then all the objects will be donwloaded to the client and the user can filter them with the *Assigned to me* button.

9. Under the *Data Filter* tab, select *PARTNER_FUNCTION*, then *VW - Person responsible* from the Low Value drop-down list.

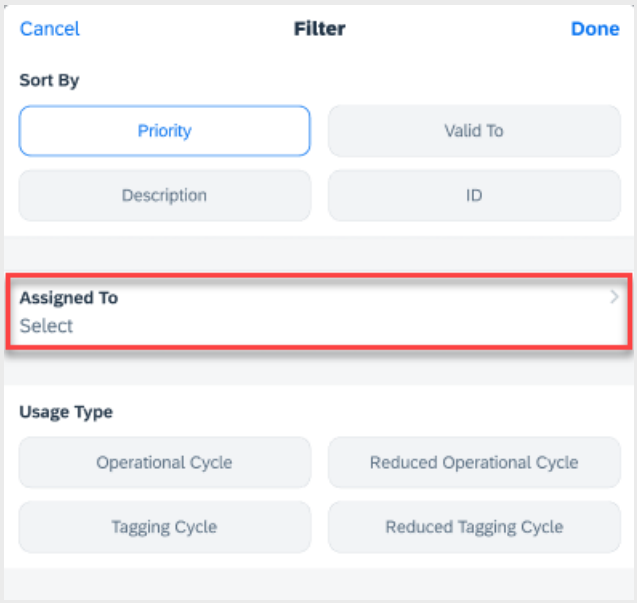


10. Click *Save* to exit Change Mode.

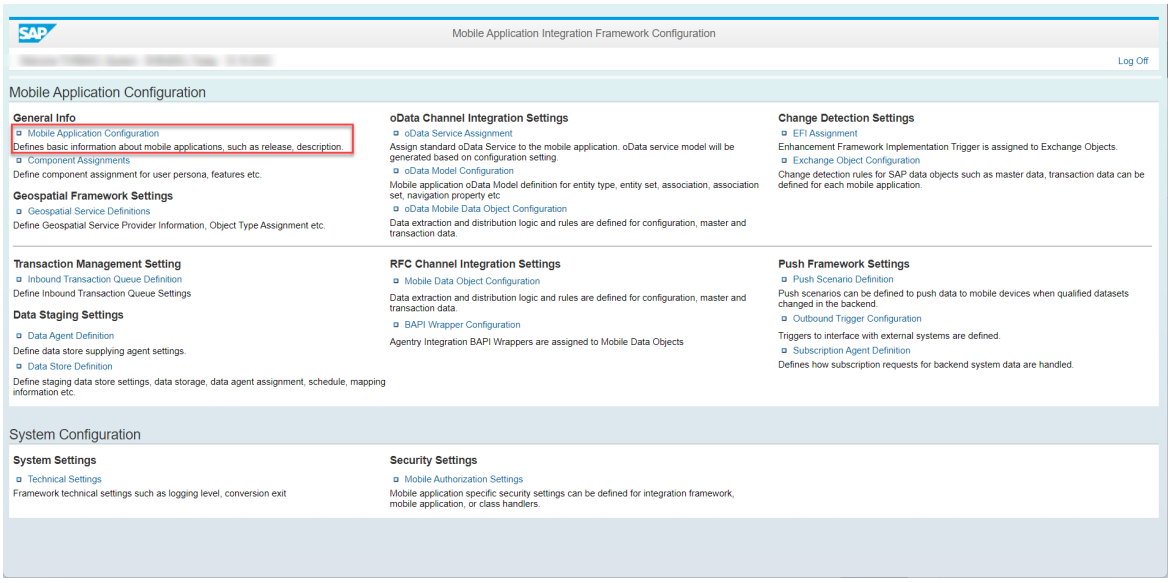
Configuring the ShowAssignedToList parameter

> ⓘ Note
>
> The following steps are optional, perform them if you want to enable for Safety Technicians to filter not just their assigned objects, but the assigned objects of other specific users as well. This filtering option will be

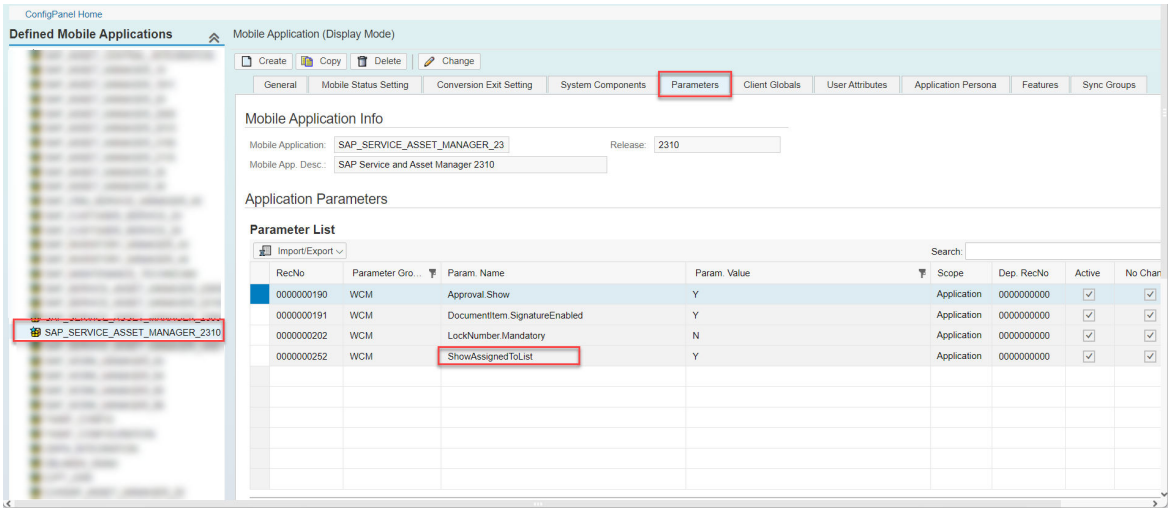
available in the *Filter* menu, from the *Assigned To* drop-down list. For more information, see Filtering Your Assigned Objects.



11. In the Configuration Panel, navigate to *Mobile Application Configuration*.



12. Select the application from the list, go to *Parameters* tab and click the *ShowAssignedToList* parameter.



13. Enter Change Mode and set the *Parameter Value* to *Y*.

> ### Note
>
> - If the *Parameter Value* is *Y*, the user will see the *Assigned To* filtering option (filtering for other users as well).
> - If the *Parameter Value* is *N*, the user will see the *Assigned To Me* filtering option (filtering only own assigned objects).

14. Click *Save* to exit Change Mode.

7.7.2 Assignment Configuration by Assignment Type 5 - Selection Variant

Context

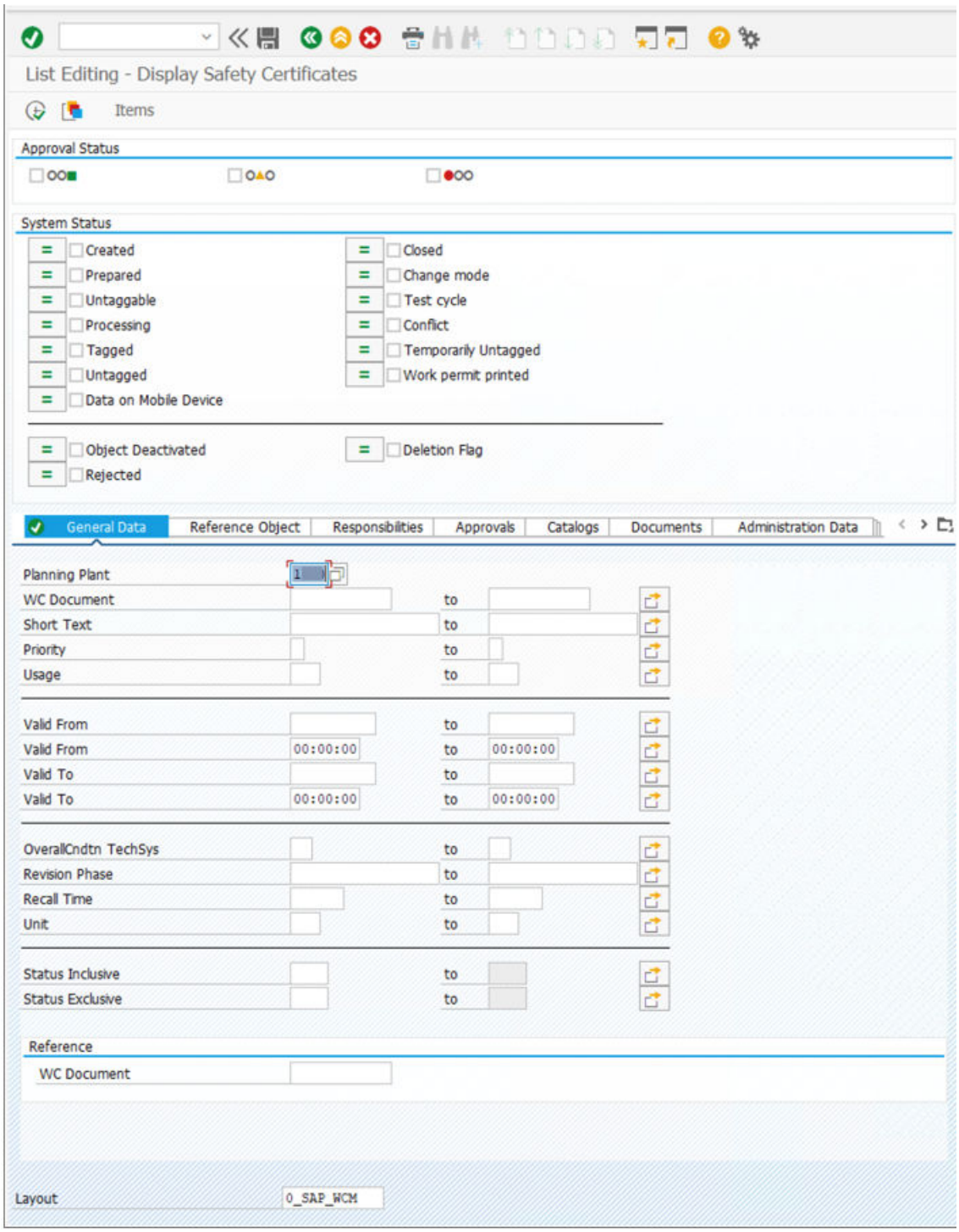
To enable the *Assigned to me* filtering option for users by the *5 - Selection Variant* Assignment Type, perform the following steps:

> ### Note
>
> Regarding the order of steps, it does not matter if you perform the SAP GUI or Configuration Panel changes first.

Procedure

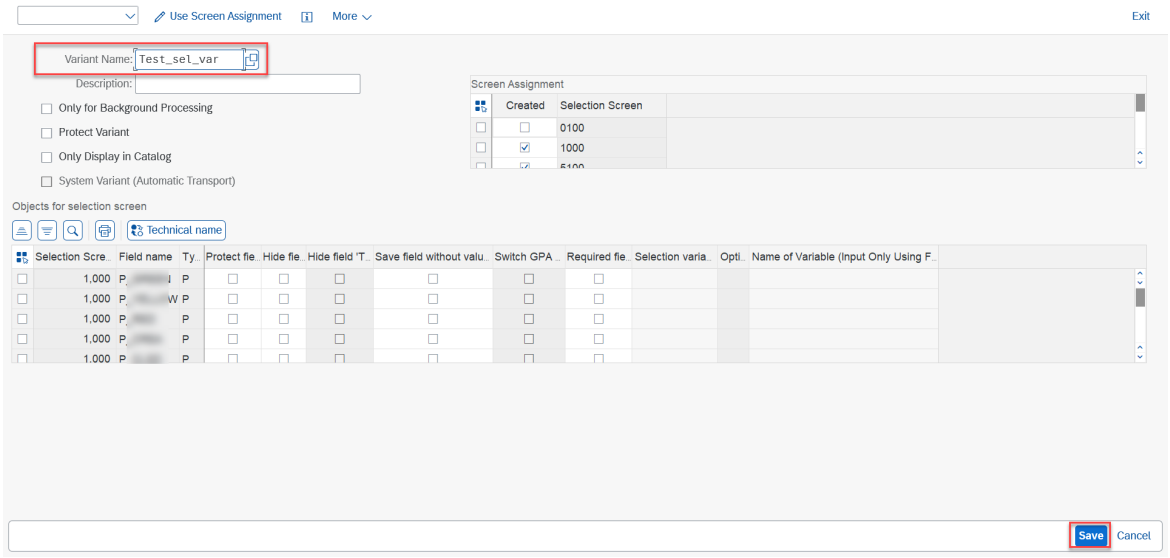
SAP GUI changes

1. To create the Selection Variant, use the `WCL4` transaction to open the *List Editing - Display Safety Cerificates* screen.

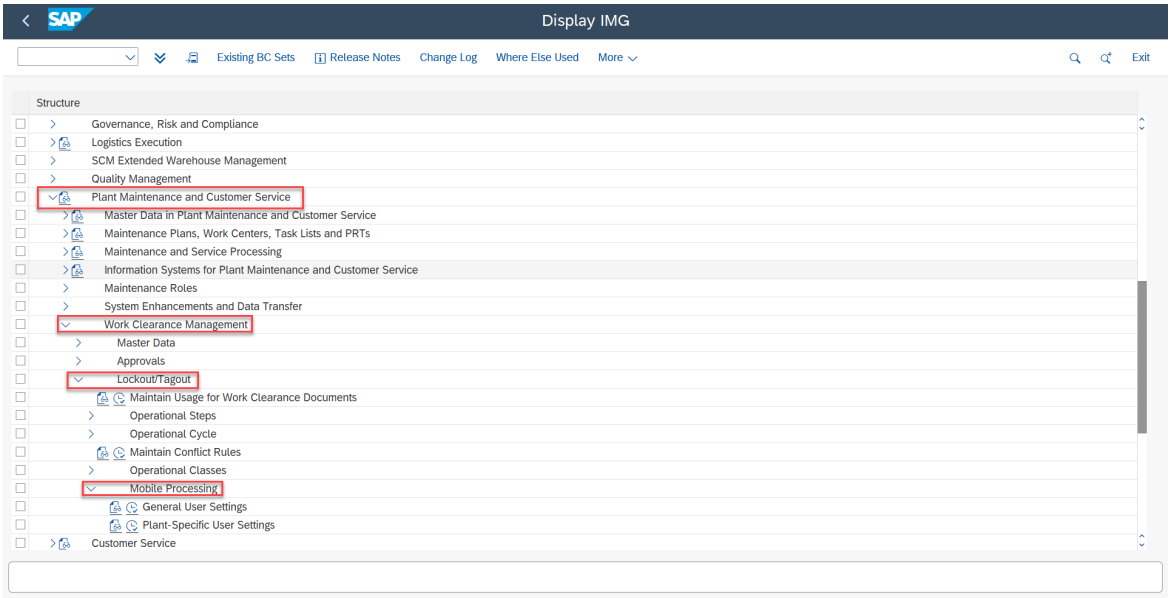


2. Select the filters you would like to use and click *Save*.

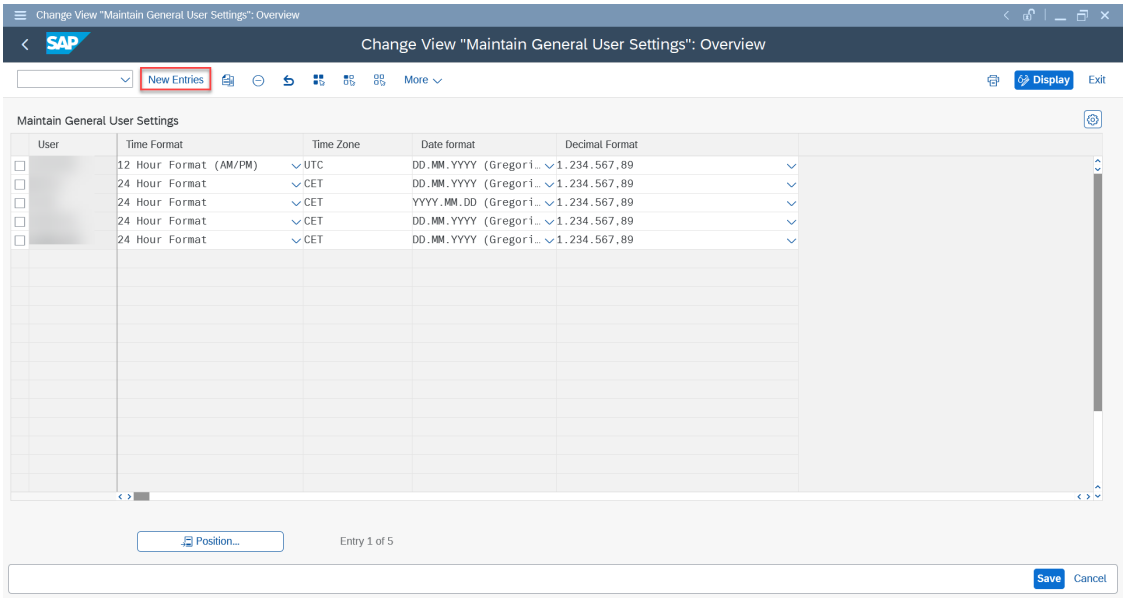
3. Enter a name for the Selection Variant and click *Save*.



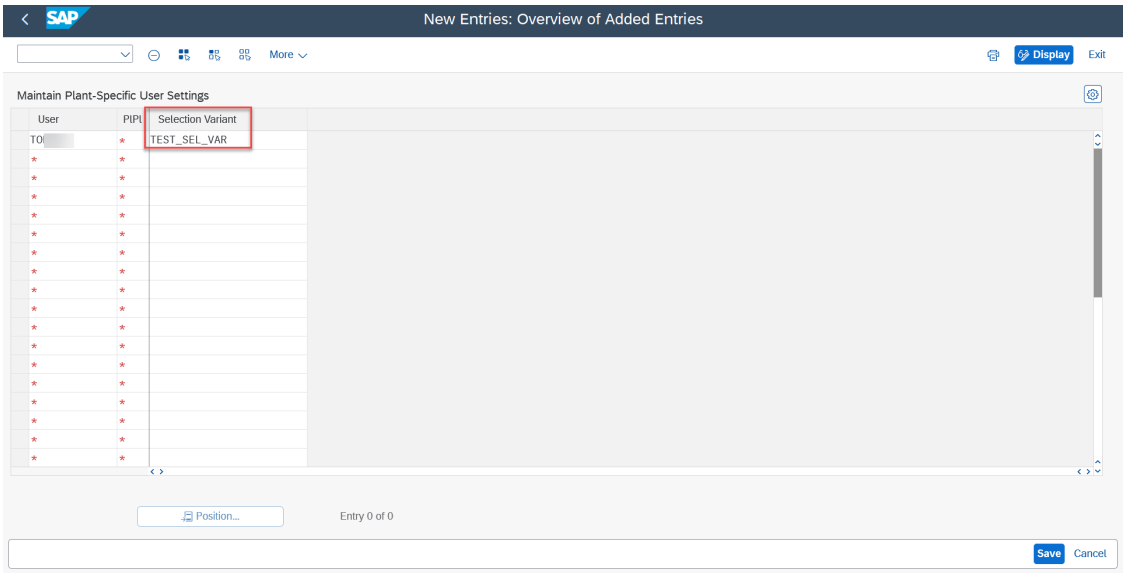
4. Go to the SPRO transaction and navigate to `Plant Maintenance and Customer Service` > `Work Clearance Management` > `Lockout/Tagout` > `Mobile Processing`.



1. Go to the `General User Settings` and add the user to the list.

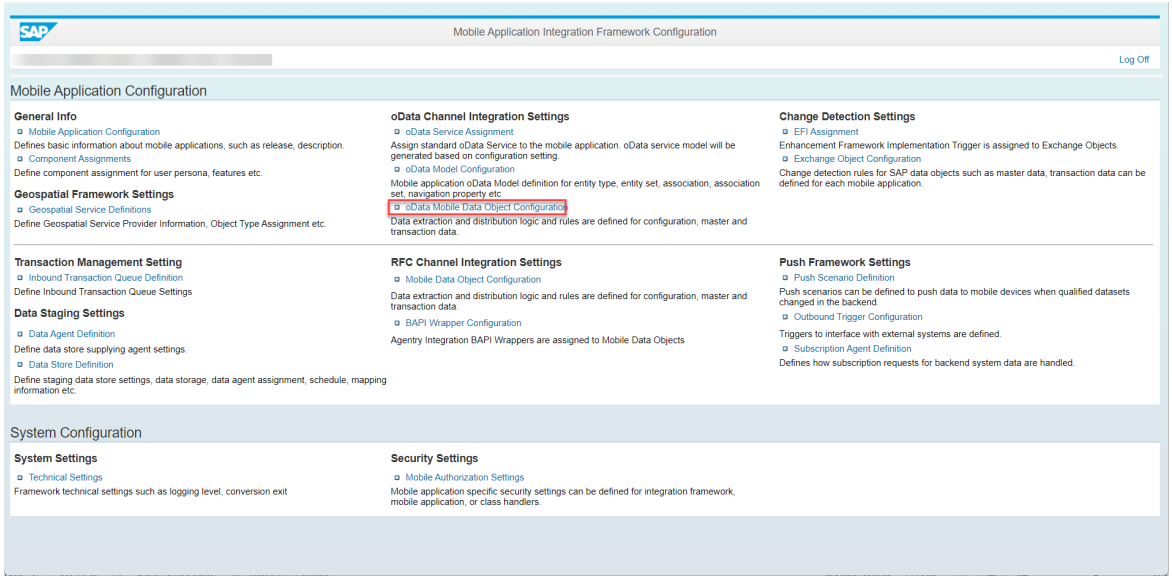


2. Go to the `Plant-Specific User Settings`, add the user to the list and assign the previously created Selection Variant to the user.

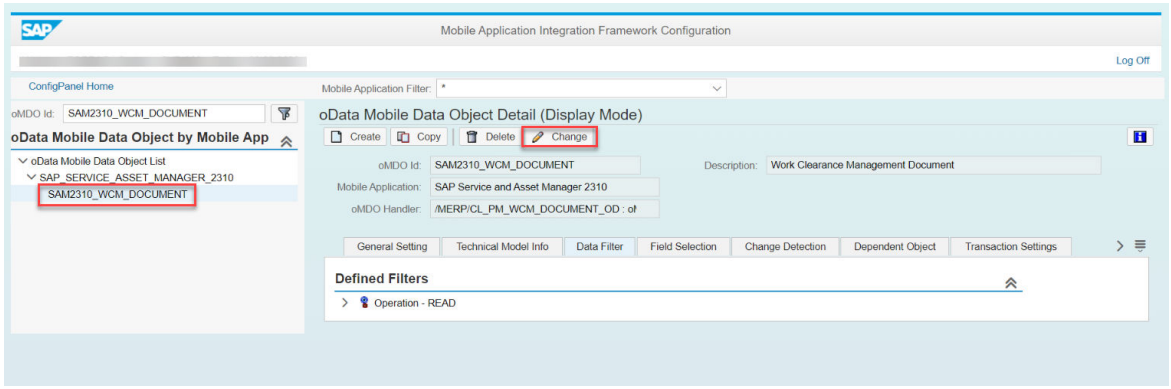


Configuration Panel changes

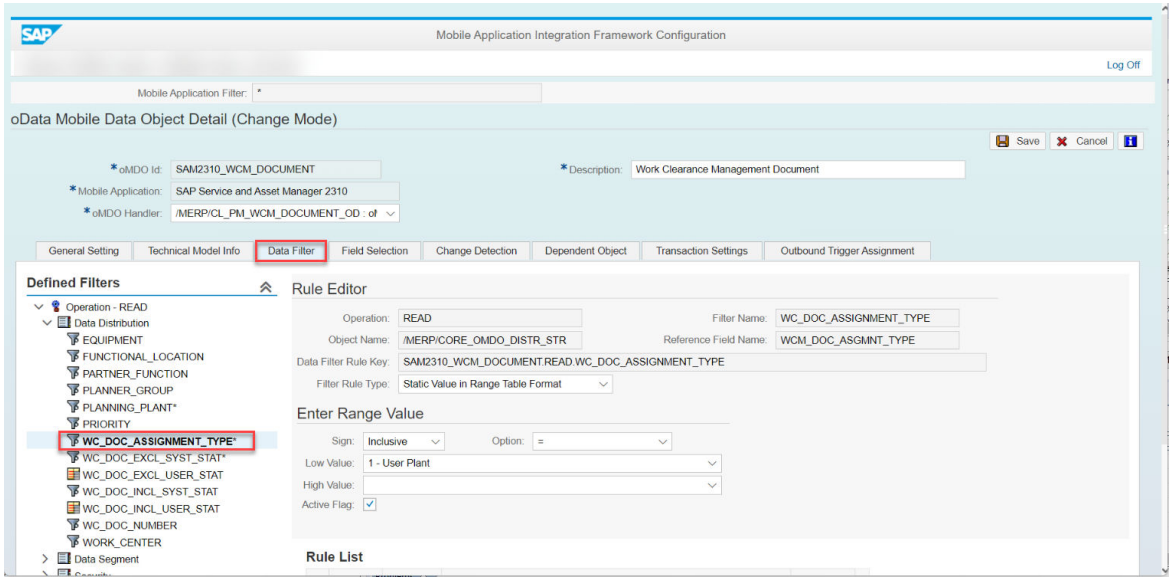
5. In the *SAP GUI*, navigate to the `/n/syclo/configpanel` transaction to open the configuration panel.
6. Under *oData Channel Integration Settings*, navigate to *oData Mobile Data Configuration.*



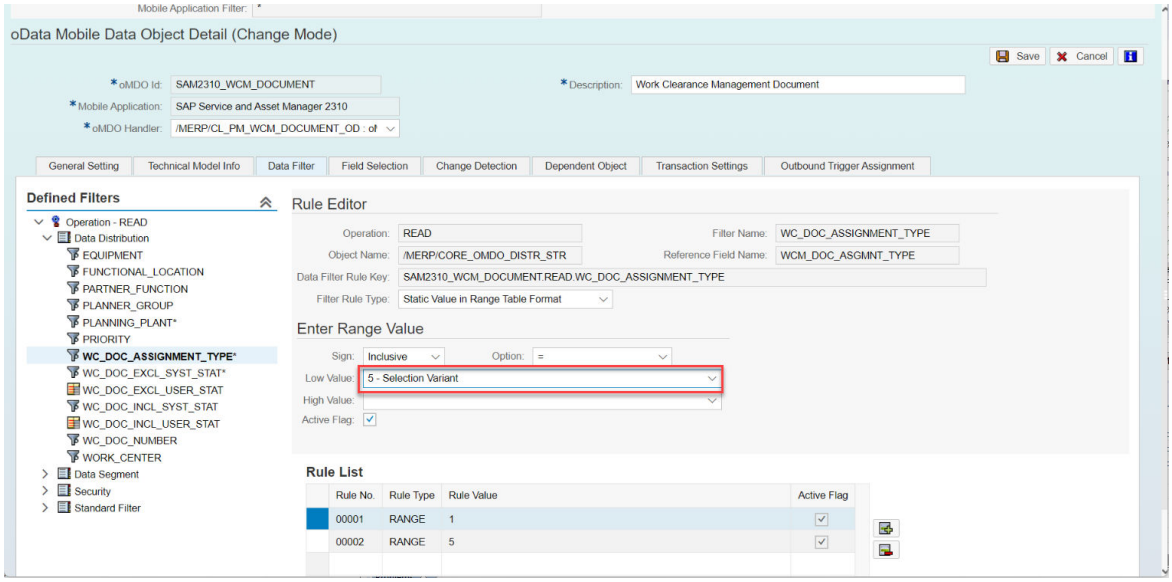
7. From the list of oMDOs, select *SAM2310_WCM_DOCUMENT* and click *Change*.



8. Under the *Data Filter* tab, select *WC_DOC_ASSIGNMENT_TYPE* from the list of filters.



9. From the Low Value drop-down list, choose *5 - Selection Variant* as assignment type.



> **Note**
>
> Consider the following in the *Rule List* table, *Rule Value* column:
>
> - If there is only one row with assignment type *5*, then only the objects assigned to the user will be downloaded to the client (*Assigned to me* button will not be displayed).
> - If there are two rows with assignment types *1* and *5* (see the screenshot above), then all the objects will be donwloaded to the client and the user can filter them with the *Assigned to me* button.

10. Click *Save* to exit Change Mode.

7.7.3 Assignment Configuration by Assignment Type 6 - Operational List + Selection Variant

Context

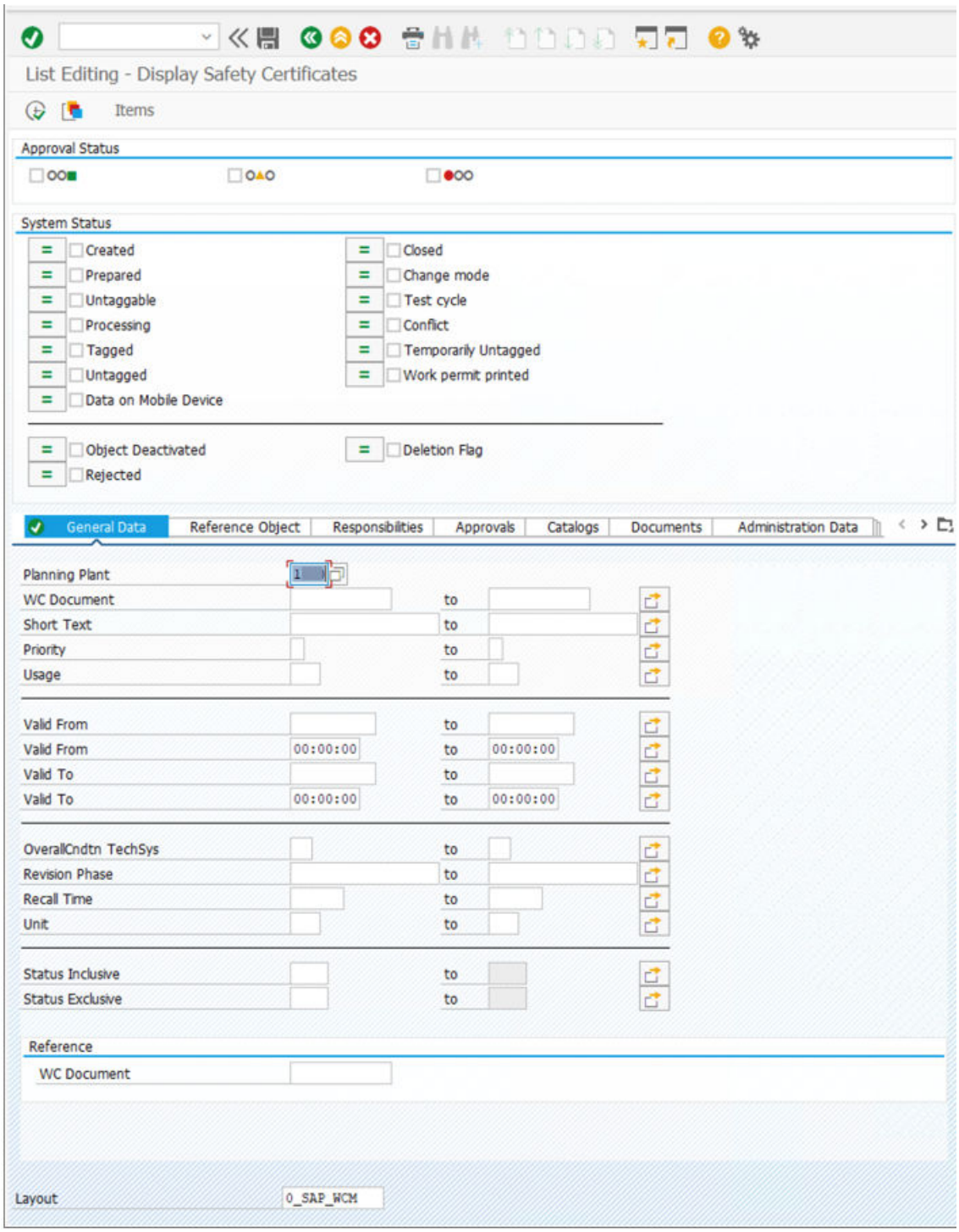
To enable the *Assigned to me* filtering option for users by the *6 - Operational List (Operational list + Selection Variant)* Assignment Type, perform the following steps:

> **Note**
>
> Regarding the order of steps, it does not matter if you perform the SAP GUI or Configuration Panel changes first.

Procedure

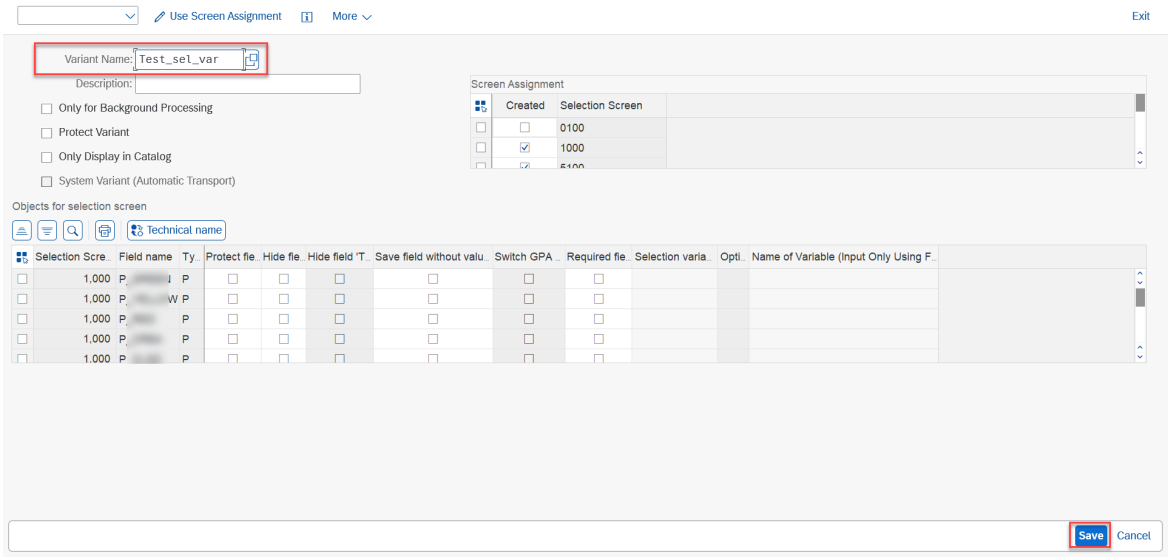
SAP GUI changes

1. To create the Selection Variant, use the `WCL4` transaction to open the *List Editing - Display Safety Cerificates* screen.

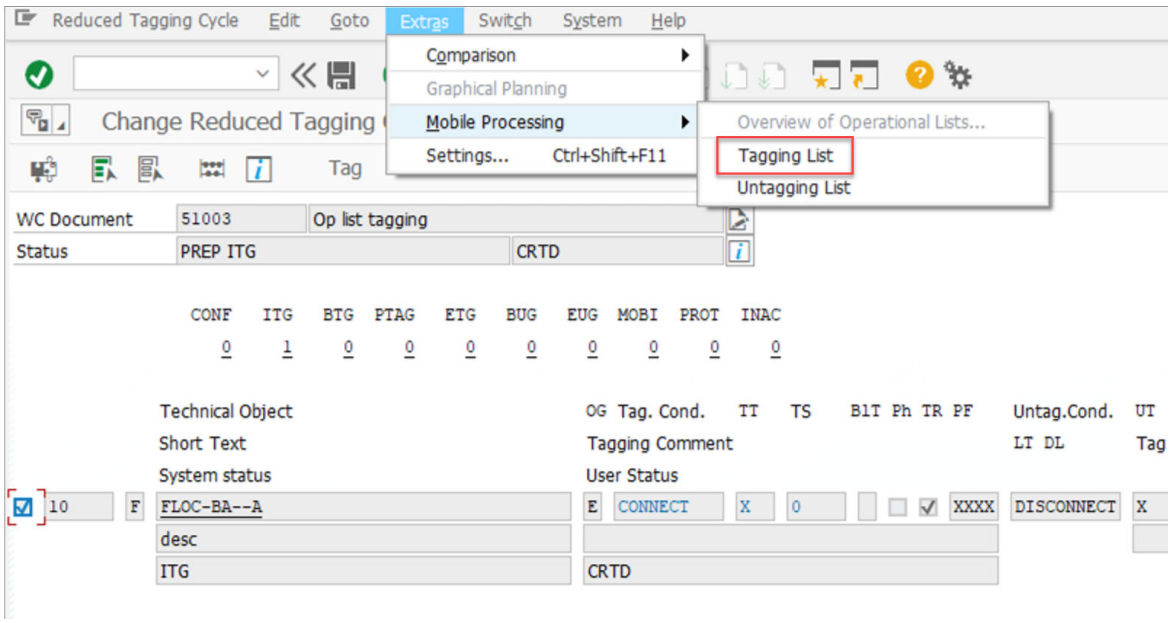


2. Select the filters you would like to use and click *Save*.

3. Enter a name for the Selection Variant and click *Save*.

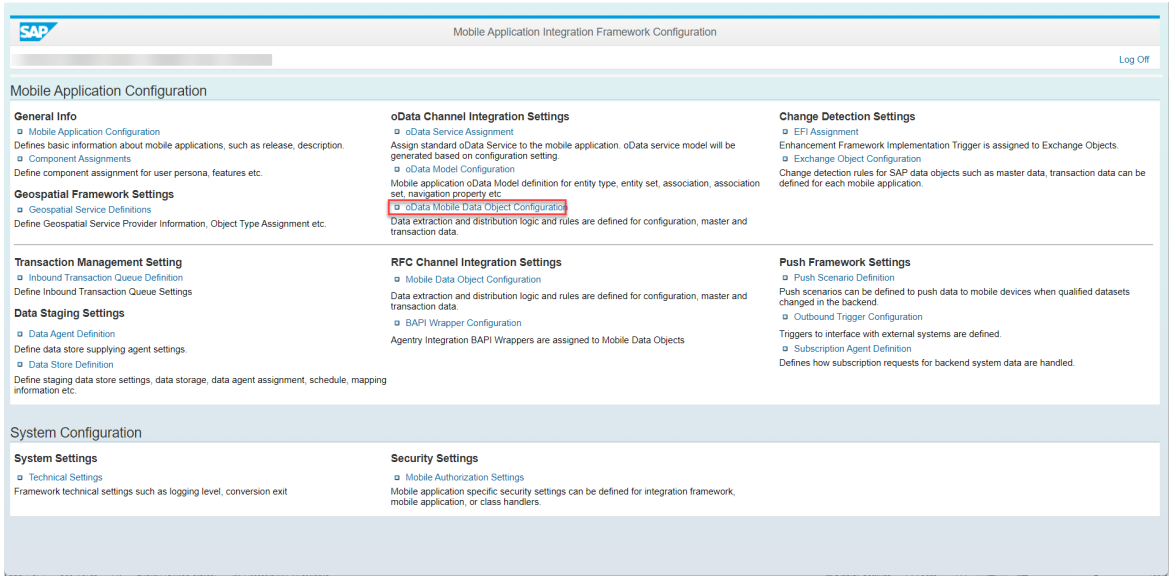


4. Enable Mobile Processing for the created Safety Certificate's Usage Type.
5. Add the Operational Item to the *Tagging List*.

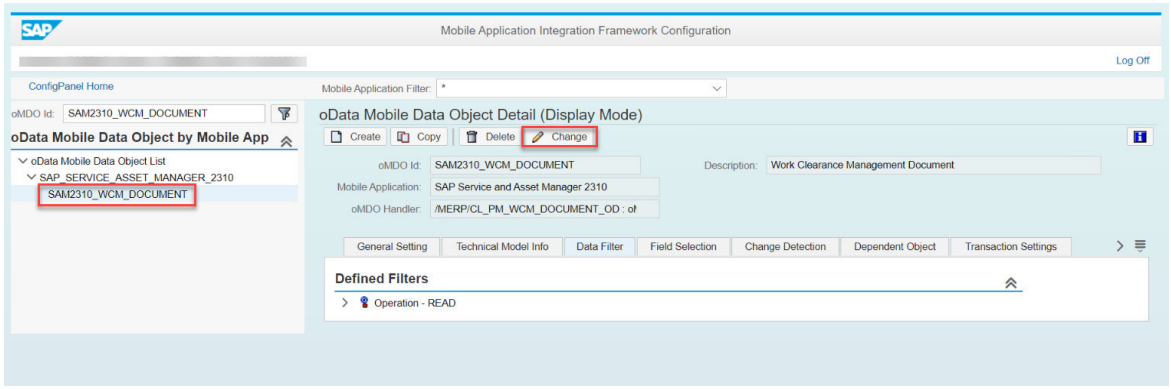


Configuration Panel changes

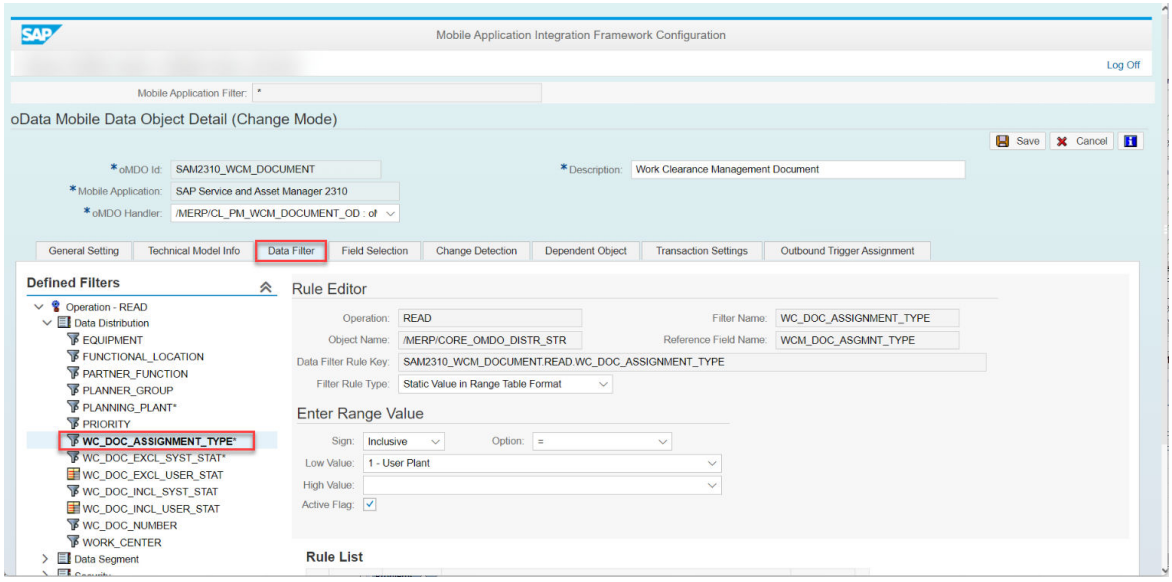
6. In the *SAP GUI*, navigate to the `/n/syclo/configpanel` transaction to open the configuration panel.
7. Under *oData Channel Integration Settings*, navigate to *oData Mobile Data Configuration*.



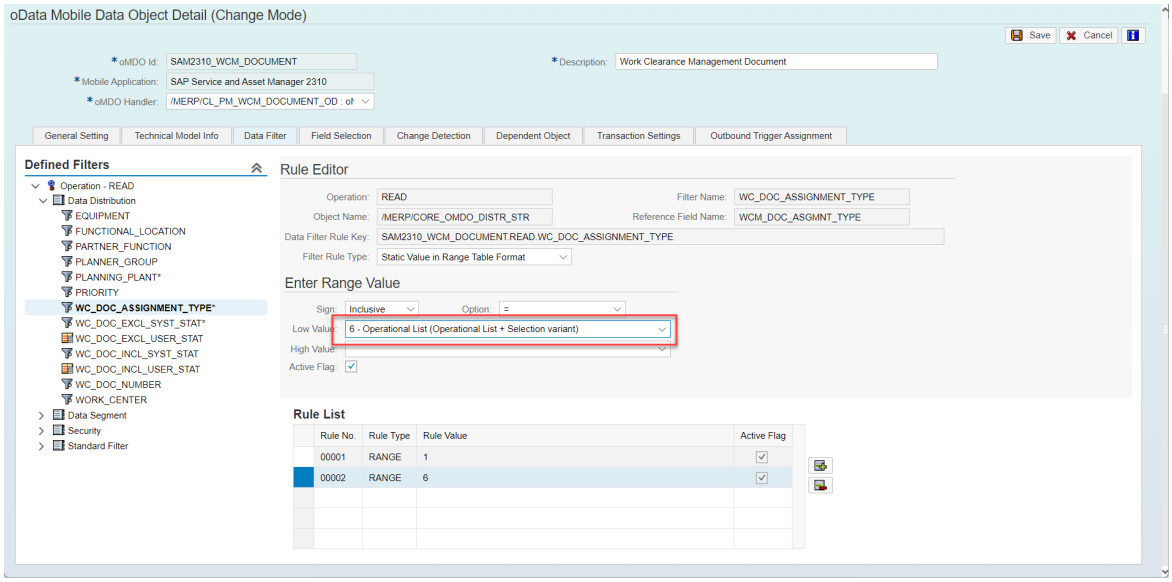
8. From the list of oMDOs, select *SAM2310_WCM_DOCUMENT* and click *Change*.



9. Under the *Data Filter* tab, select *WC_DOC_ASSIGNMENT_TYPE* from the list of filters.



10. From the Low Value drop-down list, choose *6 - Operational List (Operational list + Selection Variant)* as assignment type.



> **Note**
>
> Consider the following in the *Rule List* table, *Rule Value* column:
>
> - If there is only one row with assignment type *6*, then only the objects assigned to the user will be downloaded to the client (*Assigned to me* button will not be displayed).
> - If there are two rows with assignment types *1* and *6* (see the screenshot above), then all the objects will be donwloaded to the client and the user can filter them with the *Assigned to me* button.

11. Click *Save* to exit Change Mode.

7.8 Work Permit Approvals

Context

Safety Technicians can issue approvals on the details screen of a selected Work Permit (see Issuing Work Permit Approvals). To configure this feature for the Safety Technician persona, perform the following steps:

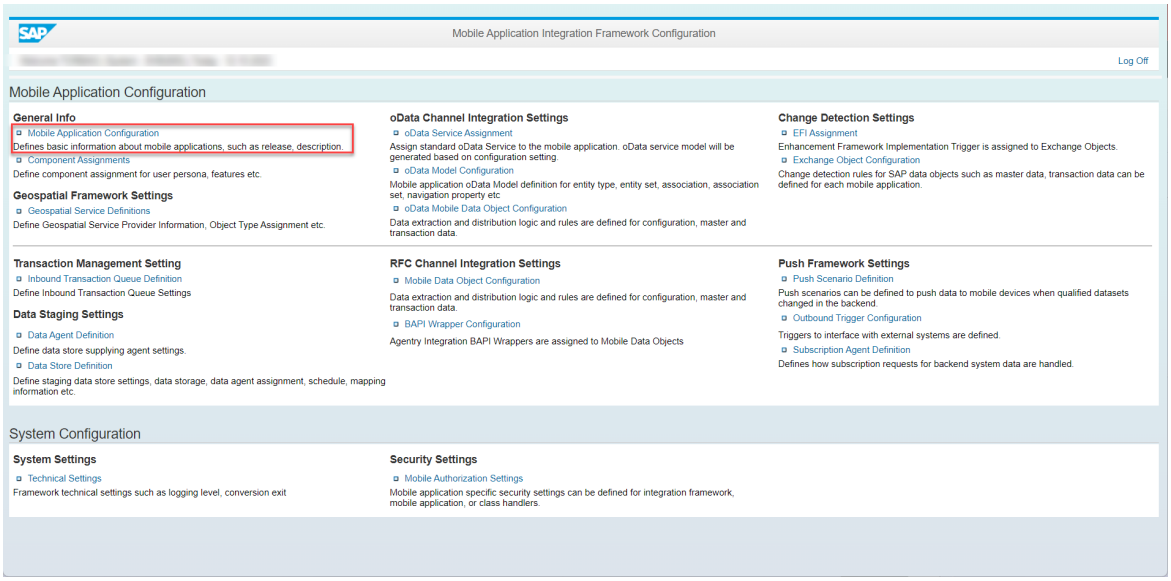
> **Note**
>
> - Regarding the order of steps, it does not matter if you perform the SPRO or Mobile Application parameter (*WCMApprovalRole*) configuration first.
> - With the Mobile Application parameter configuration you can check if the user is authorized to do **any** approvals.

- With the SPRO configuration you can check if the user is authorized to do a **specific** approval.

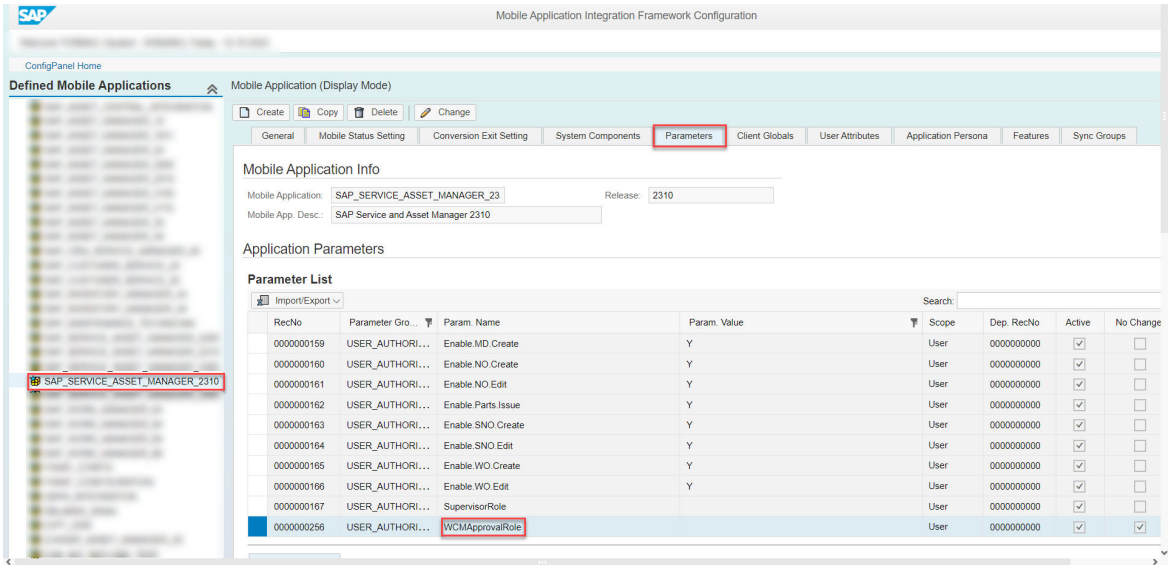
Procedure

Configuration Panel changes

1. In the *SAP GUI*, navigate to the `/n/syclo/configpanel` transaction to open the Configuration Panel.
2. Under *General Info*, navigate to *Mobile Application Configuration*.



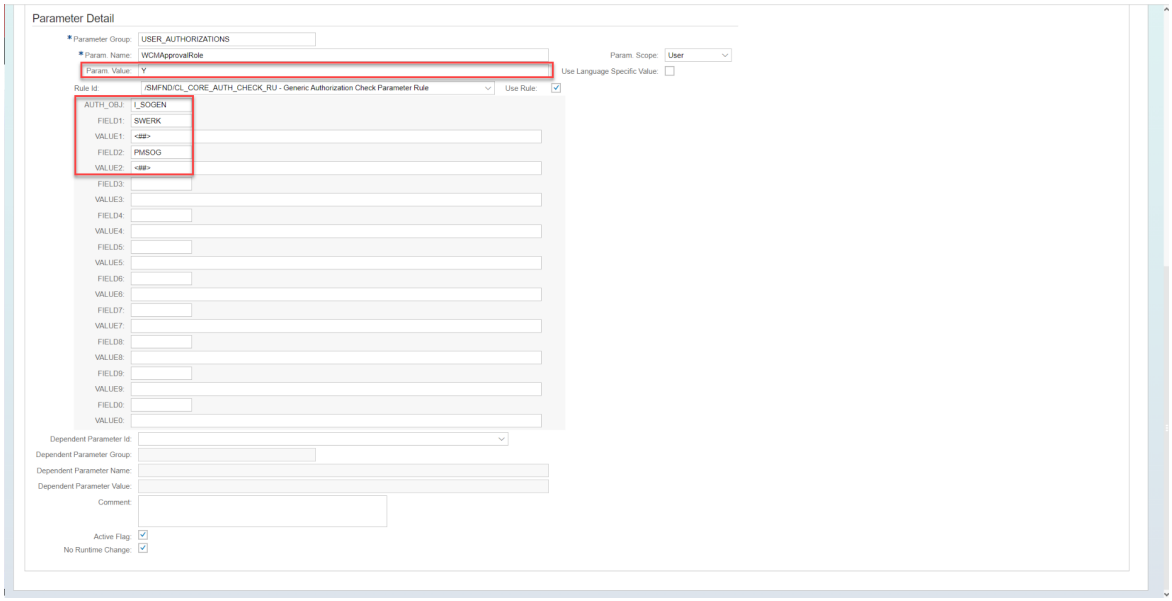
3. Select the application from the list, go to *Parameters* tab and click the *WCMApprovalRole* parameter.



> **Note**
>
> The *WCMApprovalRole* parameter checks if the specific authorization object is assigned to the user.

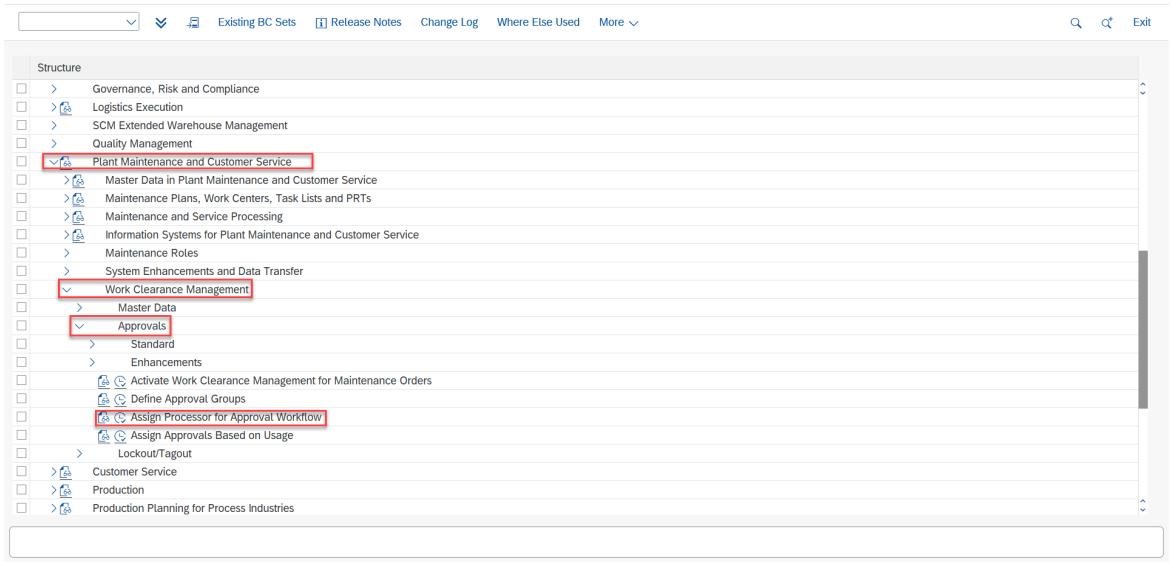
4. Under *Parameter Detail*:

    - The *AUTH_OBJ* is **I_SOGEN** by default.
    - Check if the *SWERK* and *PMSOG* fields have any values.
    - This parameter returns '**Y**' as Parameter Value if the authorization object is assigned to the user and '**N**' if it is not.

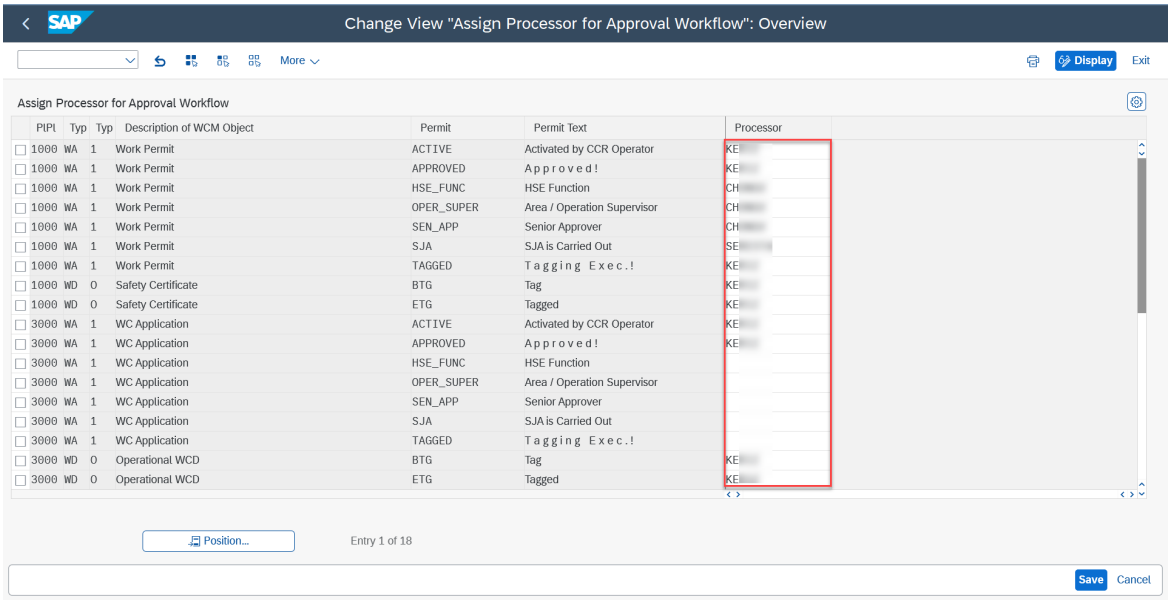


SPRO changes in SAP GUI

5. In *SAP GUI*, use the transaction code **SPRO** and go to `SAP Reference IMG > Plant Maintenance and Customer Service > Work Clearence Management > Approvals > Assign Processor for Approval Workflow`.



6. In this **SPRO** setting, you can assign a user to each plant or approval combination.
7. Assign the SAP user ID of the user in the *Processor* field.



Note

This field is a free text field, there is no validation for the SAP user ID. Please make sure you enter the ID correctly.

8 Field Service Technician Configuration on a Customer Service Back End

The Field Service Management solution connects and enables operations while simplifying and automating processes, helping to accelerate execution, improve the productivity of service teams, and control costs.

Using the Field Service Management solution, field service leaders and managers can make decisions based on real-time insights, gain visibility of field service operations, and take advantage of analytical dashboards.

Field service technicians using the SAP Service and Asset Manager app get assignment information in advance, so they are better prepared. Flexible mobile tools, including guided procedures and checklists help solve issues on the first visit while reducing time spent on administrative tasks. While on-site, technicians can collect relevant information about the assignment, get customer signatures, and sync information and back office processes quickly.

8.1 SAP Service and Asset Manager Integration with Field Service Management Scheduling

Field Service Management planning board administrators can use geolocation information from the mobile app to enable scheduling and dispatching based on the location of the technician.

Supported Scenarios

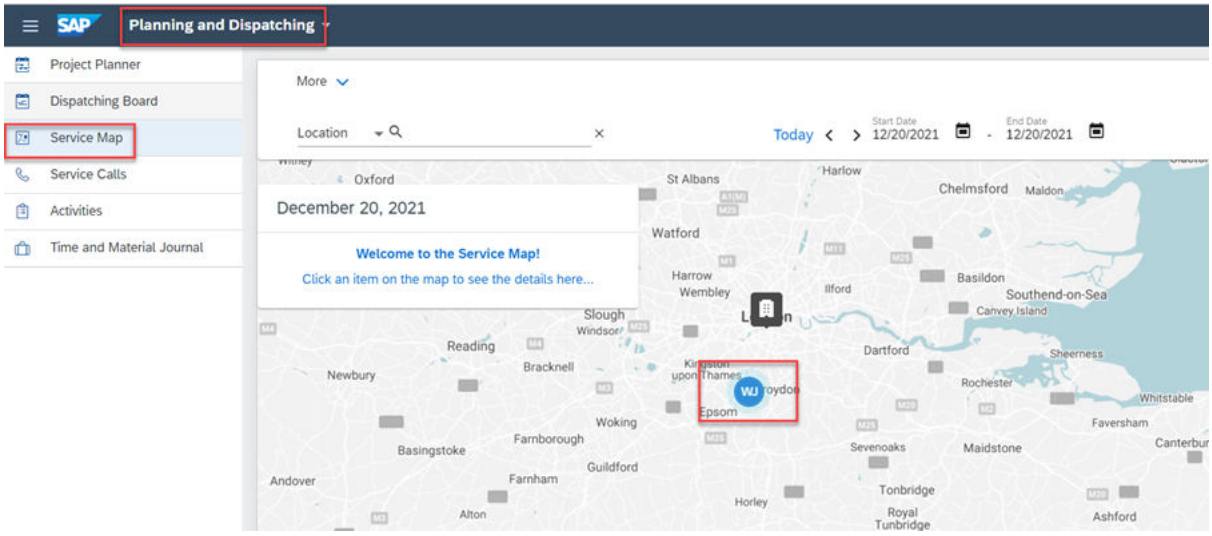
The following scenarios are supported for the integration of SAP Service and Asset Manager with Field Service Management Scheduling:

- Status updates: Status updates from SAP Service and Asset Manager are propagated to the Field Service Management Planning and Dispatching Tool. Since Field Service Management supports only activity level assignments, status transition is only supported for the *Assignment Type 2 – Operation Level Assignment to Personnel Number*. Only statuses assigned to the Field Service Management service workflow statuses are propagated to Field Service Management.
- Technician Location Tracking: The location of the technician captured from SAP Service and Asset Manager is updated to Field Service Management so that it can be viewed in the Field Service Management service map.
- Reject Operation: If a technician rejects an operation, the corresponding activity in Field Service Management is unassigned.
- Transfer Operation: If a technician transfers an operation from one technician to another, the corresponding activity in Field Service Management is reassigned from one technician to another.

Employee Location Update

- The employee location is captured in SAP Service and Asset Manager, and the location can be updated into Field Service Management.
- The mobile status changes from SAP Service and Asset Manager can be updated into the Field Service Management.

The location updates from the client can be viewed under the Field Service Management Map, *Planning and Dispatch* > *Service Map*.



8.1.1 Prerequisites for Configuring Field Service Management

It is assumed that the Proaxia Field Service Management connector is configured and set up to replicate the data between the SAP back end and the SAP Field Service Management system.

SAP Service and Asset Manager integration with Field Service Management requires that the following scenarios are configured and enabled within the Proaxia Field Service Management connector:

- **Employees:** Replicate employees (SAP HR Personnel numbers) from SAP ECC to Field Service Management as Persons. Use transaction *PA30* to configure *Infotype 0105/001* to map the user name to the user's personnel number replicated in Field Service Management.
- **Materials:** Replicate *Materials* in the back end as *Items* to Field Service Management. Items are needed to create equipment.
- **Equipment / Functional Locations:** Replicate *Equipment* and/or *Functional Location* to Field Service Management as *Equipment*. Equipment are technical objects assigned to service orders
- **Customers:** Replicate *Customers* to Field Service Management as *Business Partners*. Business partners are assigned to service orders.
- **Business Partners (Optional):** Replicate *Business Partners* to Field Service Management if they're needed in the customer scenario as part of order processing.
- **Orders:** Replicate *Service Orders*, *Work Orders*, and *Operations* to Field Service Management.
- **Assignments (Field Service Management):** Replicate *Assignments* from the Field Service Management system to the back end.

8.1.2 Integration of SAP Service and Asset Manager with Field Service Management Using the Proaxia Field Service Management Connector

To execute the supported integration scenarios between SAP Service and Asset Manager and Field Service Management, SAP Service and Asset Manager integrates directly with Field Service Management using the Service and Data APIs provided by Field Service Management. However, the underlying objects that need to be updated (service orders (service calls in Field Service Management), operations (activities in Field Service Management), employees (persons in Field Service Management)), are created by the Proaxia connector.

Therefore, you must implement the following prerequisites for the end-to-end scenarios to work:

- Switching on Sending externalID from the Proaxia Field Service Management Connector to the Field Service Management Solution [page 293]
- Implement BADI Methods [page 294]

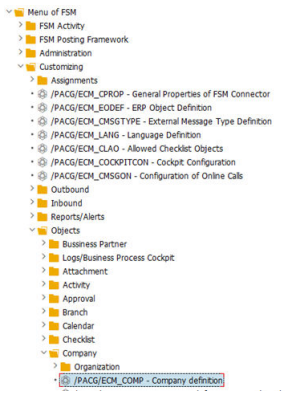
8.1.2.1 Switching on Sending externalID from the Proaxia Field Service Management Connector to the Field Service Management Solution

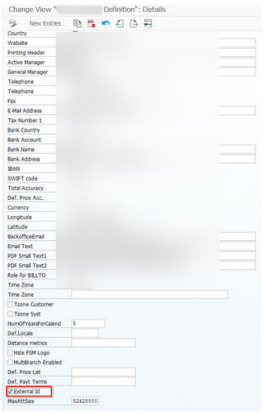
Context

> **Note**
>
> Perform these changes within the configuration of the Proaxia Field Service Management connector.

Procedure

Switch on the sending of the *externalId* from the Field Service Management connector to Field Service Management as shown in the following screenshots:





8.1.2.2 Implement BADI Methods

Context

Depending on the integration framework used, implement the */MERP/CA_FSM_CROSSREF_BADI* BADI. The methods in the BADI must return the cross-referencing information between the back end object ID and the Field Service Management object ID.

Procedure

1. Implement the local method to determine the company ID, required for the *GET_EMPLOYEE_IDS BADI* method.

```
METHOD get_compid.

    DATA: lv_string1    TYPE string,
          lv_string2    TYPE string,
          lv_account    TYPE string,
          lv_fsmaccount TYPE /pacg/ecm_cloudaccount,
          lv_fsmcompany TYPE /pacg/ecm_cloudid,
          lv_compid     TYPE /pacg/ecm_compid,
          ls_cacc       TYPE /pacg/ecm_cacc.

    lv_account = iv_account.

    SPLIT lv_account AT '&account=' INTO lv_string1 lv_string2.
    SPLIT lv_string2 AT '&company=' INTO lv_string1 lv_string2.

    lv_fsmaccount = lv_string1.
    lv_fsmcompany = lv_string2.


    ls_cacc = /pacg/ecm_cl_d_access=>get_cloudcomp_definition(
```

```
iv_cloudaccount = lv_fsmaccount
iv_cloudid      = lv_fsmcompany ).

rv_compid = ls_cacc-compid.

ENDMETHOD
```

2. Implement the method */MERP/IF_CA_FSM_CROSSREF_BADI~GET_EMPLOYEE_IDS*:

   This method is a prerequisite for saving geolocations captured from SAP Service and Asset Manager to the Field Service Management planning and scheduling board.

   The method must return the Field Service Management internal employee ID(s) (*Field* `id` of the Field Service Management *Person* entity) based on the personnel number(s) provided.

   Use the importing parameter *IO_ACI_SERVICE* to execute a service call to Field Service Management for retrieving the `id` based on the `code` in the following code example.

   > **Note**
   >
   > For simplicity, it is assumed that only a single personnel number is passed as a parameter. Adjust the code accordingly to handle multiple parameters. The following code is an example. You may need to make additional adjustments for your environment.

```
METHOD /merp/if_ca_fsm_crossref_badi~get_employee_ids.

    TYPES: BEGIN OF ty_error,
             error TYPE string,
           END OF ty_error.

    TYPES: BEGIN OF ty_fsmitem,
             id         TYPE string,
             externalid TYPE string,
             username   TYPE string,
           END OF ty_fsmitem.

    TYPES: BEGIN OF ty_persondata,
             person TYPE ty_fsmitem,
           END OF ty_persondata.

    DATA:
      lv_pernr_query TYPE string,
      ls_oblnk       TYPE /smfnd/sync_d_oblnk_h_upd_str,
      lv_id          TYPE /smfnd/sync_object_key_dte,
      lv_url         TYPE string,
      lv_char_code   TYPE n LENGTH 3,
      lt_persondata  TYPE STANDARD TABLE OF ty_persondata,
      ls_persondata  LIKE LINE OF lt_persondata,
      lt_error       TYPE STANDARD TABLE OF ty_error,
      ls_error       LIKE LINE OF lt_error,
      lv_status_code TYPE i,
      lv_reason      TYPE string,
      lv_result      TYPE string,
      lv_raw_data    TYPE xstring,
      lv_compid      TYPE /pacg/ecm_compid,
      lv_cloud_uname TYPE /pacg/ecm_cloud_uname,
      ls_pernr       LIKE LINE OF it_pernr.

    TYPES: BEGIN OF ty_personresp,
             data LIKE lt_persondata,
           END OF ty_personresp.

    DATA: ls_personresp TYPE ty_personresp.

    IF io_aci_service IS BOUND.
```

```
      IF it_pernr IS INITIAL.
        RETURN.
      ENDIF.

      lv_compid = get_compid( iv_account ).

      LOOP AT it_pernr INTO ls_pernr.

        CLEAR lv_pernr_query.

        IF ls_pernr-low IS NOT INITIAL.
          TRY.
              lv_cloud_uname = /pacg/
ecm_cl_x_util=>build_cloud_uname( iv_pernr       = ls_pernr-low
                                                                    iv_comp
id      = lv_compid ).
            CATCH /pacg/ecm_cx_main.
              "No FSM user exists
              CONTINUE.
          ENDTRY.
          CONCATENATE lv_pernr_query 'externalId="' ls_pernr-
low '"' INTO lv_pernr_query.
        ENDIF.

        IF lv_pernr_query IS NOT INITIAL.
          CONCATENATE '/api/data/v4/Person?
dtos=Person.24' '&query=' lv_pernr_query
                      '&fields=id,externalId,userName' iv_account INTO lv_url.

          CALL METHOD io_aci_service->call_service
            EXPORTING
              iv_request_uri    = lv_url
              iv_request_method = 'GET'
              it_headers        = it_headers
            IMPORTING
              ev_status_code    = lv_status_code
              ev_reason         = lv_reason
              ev_result         = lv_result
              ev_raw_data       = lv_raw_data.

          lv_char_code = lv_status_code.
          IF lv_char_code CP '4*'.
            ls_error-error = lv_result.
            APPEND ls_error TO lt_error.
          ELSE.
            /aci/
cl_util_json_handler=>deserialize(  EXPORTING json =  lv_result
                                                    CHANGING  data =  ls_pers
onresp  ).
            LOOP AT ls_personresp-data INTO ls_persondata.
              lv_id = ls_persondata-person-id.
              ls_oblnk-object_type  = 'EMPLOYEE'.
              ls_oblnk-object_key  = ls_pernr-low.
              ls_oblnk-ext_object_type  = 'PERSON'.
              ls_oblnk-ext_object_key  = lv_id.
              ls_oblnk-sys_comp = 'SAM_FSM'.
              ls_oblnk-mobile_app = iv_mapp.
              APPEND ls_oblnk TO et_oblnk.
              CLEAR ls_oblnk.
            ENDLOOP.
          ENDIF.
        ENDIF.
      ENDLOOP.
    ENDIF.
  ENDMETHOD.
```

3. Implement the method */MERP/IF_CA_FSM_CROSSREF_BADI~GET_ACTIVITY_ID*:

This method is a prerequisite for implementing the relevant status updates from SAP Service and Asset Manager to the Field Service Management planning and scheduling board. This method must return the Field Service Management internal activity ID (the `id` field of the Field Service Management *Activity* entity) based on the order and operation number provided.

Use the importing parameter *IO_ACI_SERVICE* to execute a service call to Field Service Management for retrieving the ID based on the code. See the following code example for more details.

> **Note**
>
> The following code is an example. You may need to make additional adjustments for your environment.

```
METHOD /merp/if_ca_fsm_crossref_badi~get_activity_id.

    TYPES: BEGIN OF ty_fsmitem,
             id         TYPE string,
             externalid TYPE string,
           END OF ty_fsmitem.

    TYPES: BEGIN OF ty_error,
             error TYPE string,
           END OF ty_error.

    TYPES: BEGIN OF ty_actdata,
             activity TYPE ty_fsmitem,
           END OF ty_actdata.

    DATA:
      lv_act_query   TYPE string,
      lv_id          TYPE /smfnd/sync_object_key_dte,
      lv_url         TYPE string,
      lv_char_code   TYPE n LENGTH 3,
      lt_actnr       TYPE STANDARD TABLE OF /pacg/ecm_actnr,
      lv_actnr       TYPE /pacg/ecm_actnr,
      lv_object_key  TYPE /smfnd/sync_object_key_dte,
      lt_error       TYPE STANDARD TABLE OF ty_error,
      ls_error       LIKE LINE OF lt_error,
      lt_actdata     TYPE STANDARD TABLE OF ty_actdata,
      ls_actdata     LIKE LINE OF lt_actdata,
      lv_status_code TYPE i,
      lv_reason      TYPE string,
      lv_result      TYPE string,
      lv_raw_data    TYPE xstring,
      lv_aufnr_len   TYPE i,
      lo_descr       TYPE REF TO cl_abap_elemdescr,
      ls_aufnr_dfies TYPE dfies.

    TYPES: BEGIN OF ty_actresp,
             data LIKE lt_actdata,
           END OF ty_actresp.

    DATA: ls_actresp TYPE ty_actresp.

    IF io_aci_service IS BOUND.
      IF iv_aufnr IS INITIAL OR iv_vornr IS INITIAL.
        RETURN.
      ENDIF.

      lv_object_key = iv_aufnr.
      lo_descr ?= cl_abap_elemdescr=>describe_by_data( iv_aufnr ).
      ls_aufnr_dfies = lo_descr->get_ddic_field( ).
      lv_aufnr_len = ls_aufnr_dfies-leng.
      lv_object_key+lv_aufnr_len = iv_vornr.

      "Get all activitties for order operation
```

```
      SELECT actnr INTO TABLE lt_actnr FROM /pacg/
ecm_acti WHERE aufnr = iv_aufnr AND vornr = iv_vornr.

      "Get the newest activity
      SORT lt_actnr DESCENDING.
      READ TABLE lt_actnr INTO lv_actnr INDEX 1.
      IF sy-subrc IS NOT INITIAL.
        RETURN.
      ENDIF.

      CONCATENATE lv_act_query 'externalId="' lv_actnr '"' INTO lv_act_query.

      IF lv_actnr IS NOT INITIAL.
        CONCATENATE '/api/data/v4/Activity?
dtos=Activity.39' '&query=' lv_act_query '&fields=id,externalId' iv_account IN
TO lv_url.

        CALL METHOD io_aci_service->call_service
          EXPORTING
            iv_request_uri    = lv_url
            iv_request_method = 'GET'
            it_headers        = it_headers
          IMPORTING
            ev_status_code    = lv_status_code
            ev_reason         = lv_reason
            ev_result         = lv_result
            ev_raw_data       = lv_raw_data.

        lv_char_code = lv_status_code.
        IF lv_char_code CP '4*'.
          ls_error-error = lv_result.
          APPEND ls_error TO lt_error.
        ELSE.
          /aci/
cl_util_json_handler=>deserialize(  EXPORTING json =  lv_result
                                                    CHANGING  data =  ls_actres
p  ).
          LOOP AT ls_actresp-data INTO ls_actdata.
            lv_id = ls_actdata-activity-id.
            es_oblnk-object_type  = 'OPERATION'.
            es_oblnk-object_key  = lv_object_key.
            es_oblnk-ext_object_type  = 'ACTIVITY'.
            es_oblnk-ext_object_key  = lv_id.
            es_oblnk-sys_comp = 'SAM_FSM'.
            es_oblnk-mobile_app = iv_mapp.
      ev_id = ls_actdata-activity-id.
            EXIT.
          ENDLOOP.
        ENDIF.
      ENDIF.
    ENDIF.

  ENDMETHOD.
```

4. Implement the method */MERP/IF_CA_FSM_CROSSREF_BADI~GET_SERV_ASSIGN_ID*:

   This method is a prerequisite for implementing relevant status updates from SAP Service and Asset Manager to the Field Service Management planning and scheduling board. The method must return the Field Service Management internal activity ID (the `id` field of the Field Service Management *Activity* entry) based on the order and operation number provided.

   Use the *QUERY_FSM_SERVICEASSIGN* method from the */MERP/CL_CA_FSM_INTEGRATION* class to get this value based on the following code example.

> **Note**
>
> The following code is an example. You may need to make additional adjustments for your environment.

```
METHOD /merp/if_ca_fsm_crossref_badi~get_serv_assign_id.
    DATA lref_fsm_integration TYPE REF TO /merp/cl_ca_fsm_integration.

    TRY.
        " Create FSM object
        CREATE OBJECT lref_fsm_integration
          EXPORTING
            iv_mapp = iv_mapp.
      CATCH /merp/cx_core_exception_gen INTO DATA(lref_badi_exception).
        RETURN.
    ENDTRY.
    " Fetch Service Assignment from FSM if it exists
    lref_fsm_integration->query_fsm_serviceassign(
      EXPORTING
        iv_aufnr       = iv_aufnr
        iv_vornr       = iv_vornr
        iv_activity_id = iv_activity_id
      IMPORTING
        ev_id          = ev_id
        ev_activity_id = ev_activity_id
        es_oblnk       = es_oblnk
        ).
  ENDMETHOD.
```

5. Implement the */MERP/IF_CA_FSM_CROSSREF_BADI~GET_SERV_ASSIGN_STATUS_ID*:

   This method is a prerequisite for implementing relevant status updates from SAP Service and Asset Manager to the Field Service Management planning and scheduling board. The method must return the Field Service Management internal activity ID (the `id` field of the Field Service Management *Activity* entry) based on the order and operation number provided.

   Use the *QUERY_FSM_SERVICEASSIGNSTATUS* method from the */MERP/CL_CA_FSM_INTEGRATION* class to get this value based on the following code example.

   > **Note**
   >
   > The following code is an example. You may need to make additional adjustments for your environment.

```
METHOD /merp/if_ca_fsm_crossref_badi~get_serv_assign_status_id.
    DATA lref_fsm_integration TYPE REF TO /merp/cl_ca_fsm_integration.

    TRY.
        " Create FSM object
        CREATE OBJECT lref_fsm_integration
          EXPORTING
            iv_mapp = iv_mapp.
      CATCH /merp/cx_core_exception_gen INTO DATA(lref_badi_exception).
        RETURN.
    ENDTRY.

    lref_fsm_integration->query_fsm_serviceassignstatus(
        EXPORTING
          iv_aufnr       = iv_aufnr
          iv_vornr       = iv_vornr
          iv_activity_id = iv_activity_id
        IMPORTING
          ev_id          = ev_id
          es_oblnk       = es_oblnk
          ).
```

```
ENDMETHOD.
```

6. Implement the BADI method *IS_VALID_ORDER_FOR_REPLICATION* from the BADI */MERP/CA_FSM_CROSSREF*.

   This method must return a *TRUE* (*X*) value if the supplied order ID or order type is valid for replication to Field Service Management. The status changes on the operation (activity in Field Service Management) are replicated to the Field Service Management Scheduling and Dispatching board only if a *TRUE* value is returned by the method.

   If this method isn't implemented, by default no replication of status occurs from SAP Service and Asset Manager to Field Service Management

8.1.3 Configuring Connectivity between SAP Service and Asset Manager and Field Service Management

Context

The following configuration is needed to establish a connectivity between SAP Service and Asset Manager and Field Service Management. For information on announcing root certificate change, refer to Announcing Root Certificate Change.

Procedure

1. **Create a client in Field Service Management:**
   a. Log on to your Field Service Management administrative URL.
   b. Create a new client with the following parameters:
      - **Client Authentication Method:** CLIENT_SECRET
      - **User Groups:** Admin
   c. Save the *Client ID* and *Client Secret*.
2. **Configure STRUST in the back end:**
   a. Download the *ISRG Root X1* certificate. See the Announcing Root Certificate Change topic for more details.

      > **Note**
      >
      > The `.pem` file can be downloaded.

   b. Install the *ISRG Root X1* certificate against the *SSL Client (Anonymous)* in the Trust Manager.
3. **Create RFC destinations in the back end:**
   a. Create a destination for establishing the oAuth connection between the back end and Field Service Management with the following parameters:

- **Name:** SAM2310_FSM_OAUTH
- **Connection Type:** G (HTTP Connection to External Server)
- **Technical Settings**
  - **Target Host:** auth.coresuite.com
  - **Path Prefix:** /api/oauth2/v1/token
- **Logon and Security**
  1. Select *Basic Authentication*
  2. Under *User and Password* enter the client ID and client secret from *Step 1*.
  3. Under *Security Options* specify the following:
     - **SSL:** Active
     - **SSL Certificate:** ANONYM SSL Client (Anonymous)

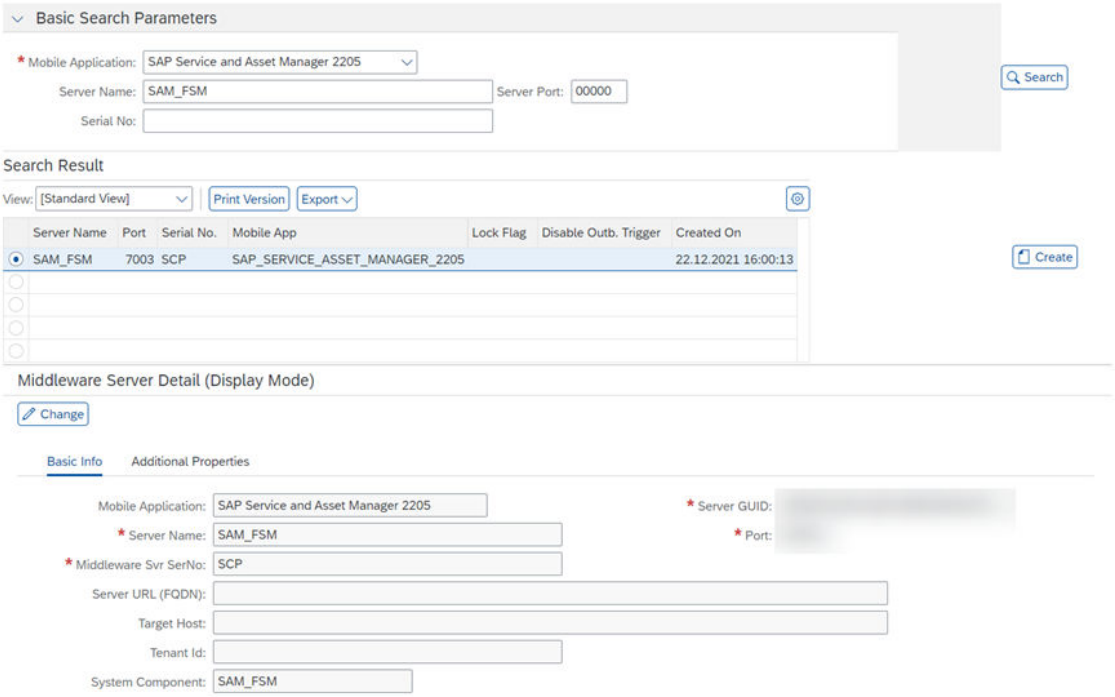
b. Create a destination for implementing the API calls to Field Service Management with the following parameters:

- **Name:** SAM2310_FSM_API
- **Connection Type:** G (HTTP Connection to External Server)
- **Technical Settings**
  - **Target Host:** <cluster e.g. us/eu>.coresuite.com
- **Logon and Security**
  1. Select *Do not use a user*.
  2. Under *Security Options*, specify the following:
     - **SSL:** Active
     - **SSL Certificate:** ANONYM SSL Client (Anonymous)

8.1.4 Configuring Server Parameters

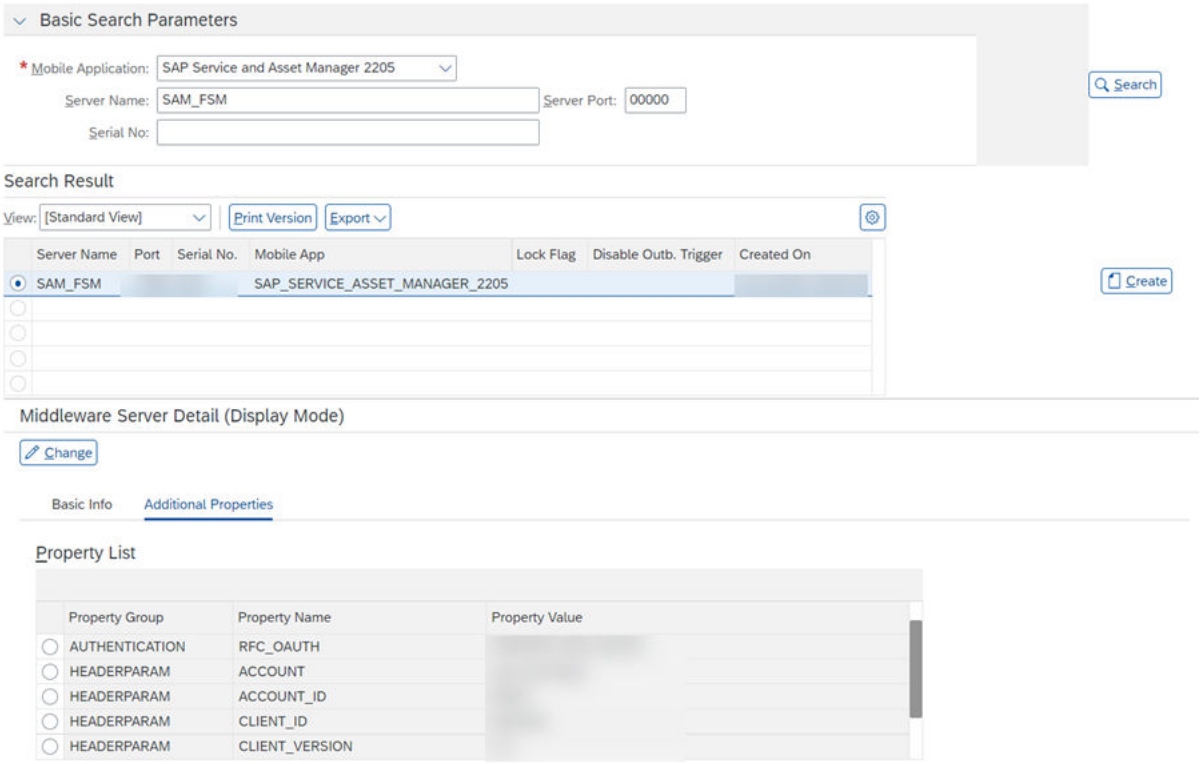
Procedure

1. Log on to the *Administration and Monitoring Portal* in the backend using transaction */SYCLO/ADMIN*. Under the *Administration* tab, select *Server Management*.
2. Create a new server with the name *SAM_FSM* for the application *SAP_SERVICE_ASSET_MANAGER_<version>*.



3. Under *Server Properties* maintain the following properties:

Property Group	Property Name	Property Value
AUTHENTICATION	RFC_OAUTH	Name of the RFC in *Step 3.1* of the Configuring Connectivity between SAP Service and Asset Manager and Field Service Management [page 300] procedure
AUTHENTICATION	RFC_API	Name of the RFC in *Step 3.2* of the Configuring Connectivity between SAP Service and Asset Manager and Field Service Management [page 300] procedure
HEADERPARAM	ACCOUNT	Field Service Management Account Name
HEADERPARAM	ACCOUNT_ID	Field Service Management Account ID
HEADERPARAM	COMPANY	Field Service Management Company Name
HEADERPARAM	COMPANY_ID	Field Service Management Company ID
HEADERPARAM	CLIENT_ID	Client ID to send to Field Service Management (e.g: SYSid.Client)
HEADERPARAM	CLIENT_VERSION	1.0



> **Note**
>
> The following additional server parameters can be defined if needed but they are optional. Their default values are also specified below. Changing these values may result in the need for customers to customize their backend integration code.

Property Group	Property Name	Property Value	Description
AUTHENTICATION	AUTH_TYPE	OAUTH	Authentication type used to communicate with Field Service Management – Default is OAUTH.
DTO	STATUSDEF	ServiceAssignmentStatus-Definition.9	Field Service Management Data Transfer Object (DTO) version. Can be changed if a newer version of a DTO is needed.
	ITEM	Item.23	
	EQUIPMENT	Equipment.23	
	BUSINESSPARTNER	BusinessPartner.23	
	ADDRESS	Address.21	
	PERSON	Person.24	
	SERVICEASSIGN	ServiceAssignment.28	
	SERVICECALL	ServiceCall.26	
	ACTIVITY	Activity.39	

Property Group	Property Name	Property Value	Description
	SERVICEASSIGNSTATUS	ServiceAssignmentStatus.15	
INTEGRATION	CREATEPERSON	admin	User for creating new objects in Field Service Management (related to SAP Service and Asset Manager only)
UI_URL_PATH	ADDRESS	/api/data/v4/Address	URI strings for the Field Service Management api call
	EQUIPMENT	/api/data/v4/Equipment	
	ITEM	/api/data/v4/Item	
	PERSON	/api/data/v4/Person	
	BUSINESSPARTNER	/api/data/v4/BusinessPartner	
	SERVICEASSIGN	/api/data/v4/ServiceAssignment	
	QUERY	/api/query/v1	
	SERVICEASSIGNSTATUS	/api/data/v4/ServiceAssignmentStatus	
	SERVICECALL	/api/data/v4/ServiceCall	
	SERVICEORDERCREATE	/service-management/api/v2/composite-tree/service-calls?autoCreateActivity=false	
	ACTIVITY	/api/data/v4/Activity	

4. *Save* your changes.

8.1.5 Configuring the Field Service Management Personas

Prerequisites

Ensure the *FIELD_SERVICE_TECHNICIAN* persona is added and activated in the Administration and Monitoring Portal.

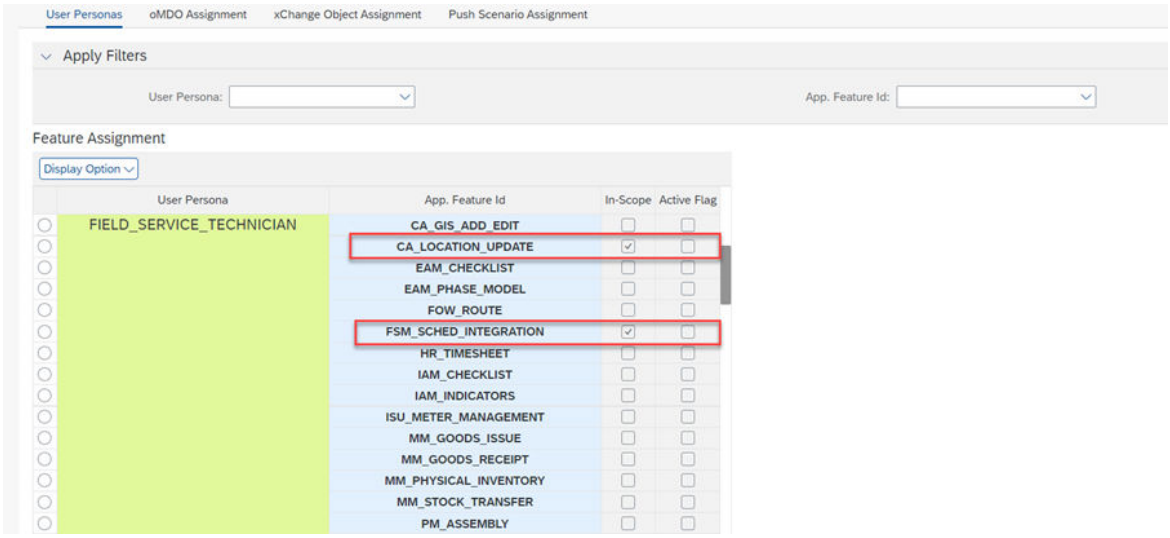
For detailed information about personas and how they work, see the Configuring Personas and Features Overview [page 83] topic and subtopics.

Context

Use the following procedure to enable the *FSM_SCHED_INTEGRATION* and *CA_LOCATION_UPDATE* features. They are disabled by default.

Procedure

1. From the ConfigPanel home screen, navigate to *Component Assignments* > *Select Mobile Application* > *User Personas tab*. You can then filter for only the *FIELD_SERVICE_TECHNICIAN* user persona, if desired.
2. Click the *Change* button.
3. Check the *Active Flag* for *CA_LOCATION_UPDATE* and *FSM_SCHED_INTEGRATION*.



4. *Save* your changes.

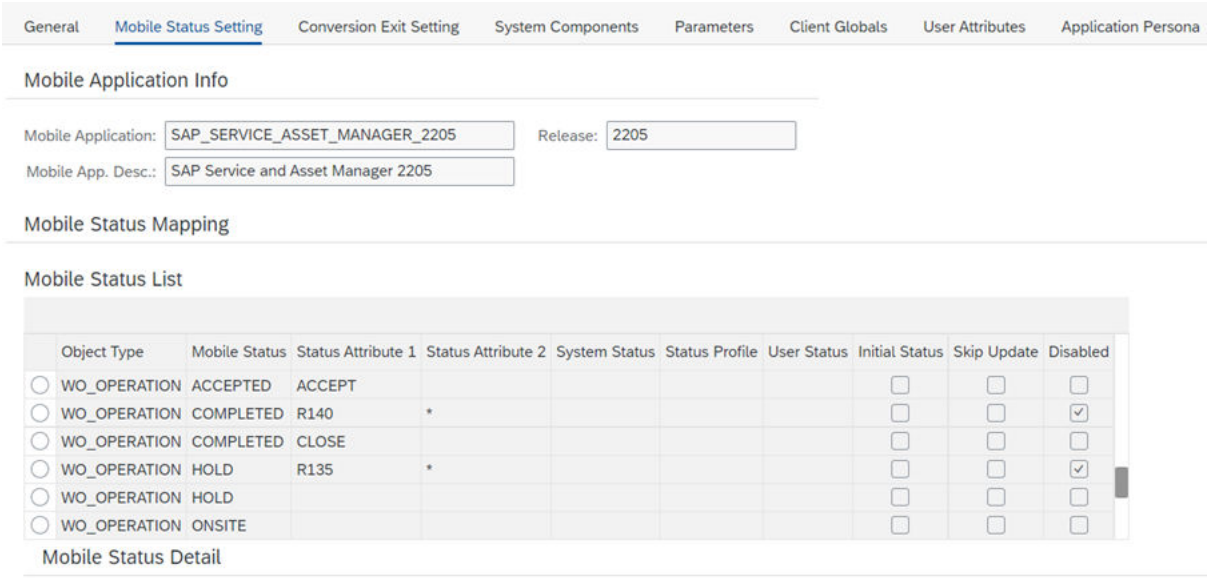
8.1.6 Status Configuration

The only assignment type supported for the Field Service Management solution is *Assignment Type 2*. Therefore, status changes are only supported at the operation level.

The following table shows the default SAP Service and Asset Manager statuses supported for the Field Service Technician persona as defined in the *Mobile Status Settings Configuration* on the ConfigPanel:

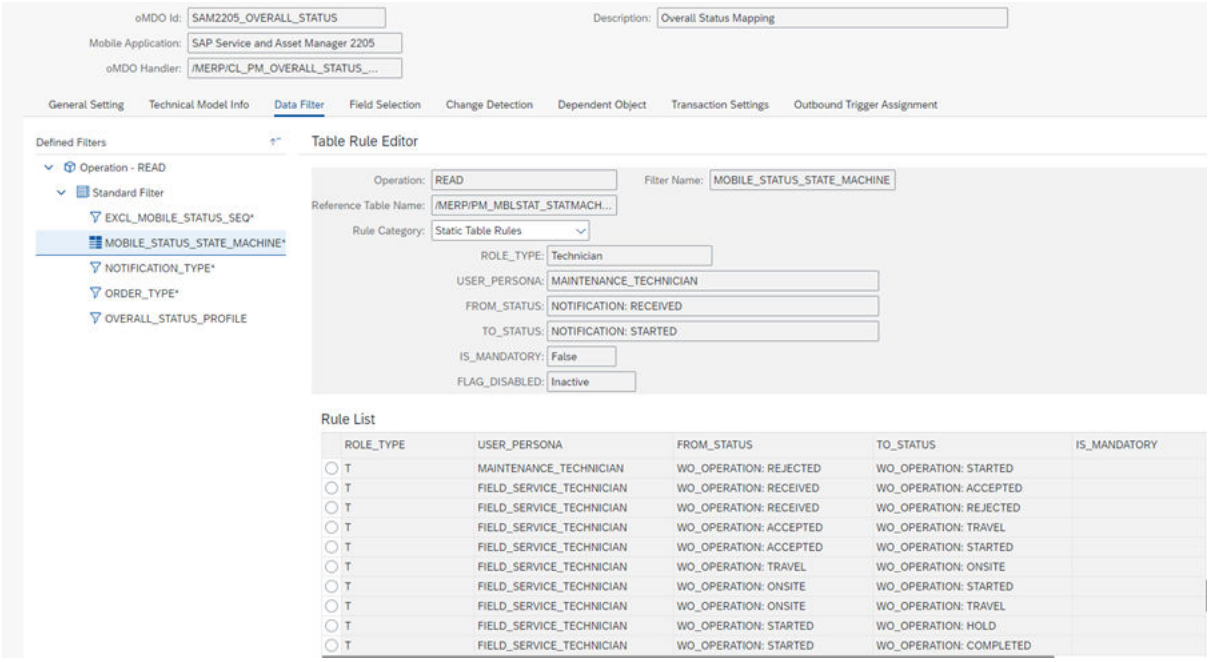
Object Type	Mobile Status	Initial Status (Y/N)	Status Attribute 1 (FSM status)
WO_OPERATION	RECEIVED	Y	new
WO_OPERATION	ACCEPTED		accept

Object Type	Mobile Status	Initial Status (Y/N)	Status Attribute 1 (FSM status)
WO_OPERATION	REJECTED		
WO_OPERATION	TRAVEL		travel
WO_OPERATION	ONSITE		
WO_OPERATION	STARTED		work
WO_OPERATION	HOLD		
WO_OPERATION	COMPLETED		close
WO_OPERATION	TRANSFER	No status update	



If any additional SAP Service and Asset Manager statuses are used or a custom service workflow is used in Field Service Management – the additional statuses can be configured in the *Mobile Application* > *Mobile Status Settings Configuration*. In this case, customization may be needed at the backend and/or MDK level depending on how the statuses need to be used.

The default status transitions are defined in the **SAM2310_OVERALL_STATUS** oMDO, *MOBILE_STATUS_STATE_MACHINE* filter. If additional status transitions are needed or existing ones need to be changed, then the filter values in this oMDO need to be changed. In this case, customization may be needed at the backend and/or MDK level depending on how the status transitions work.



8.2 Expense and Mileage Configuration

SAP Service and Asset Manager supports expense and mileage entries for the Field Service Technician persona.

The configuration of separate *Activity Types* and *Work Center* combinations are required to be able to enter expenses and mileage using SAP Service and Asset Manager. Expenses and mileage are entered as *IW41* confirmations in the back end. Both planned and unplanned mileage recordings are supported. Since the Expense and Mileage entries are stored in the backend as confirmations, due to the limitation of the corresponding field in the backend, a single decimal place is supported for the Expense and Mileage values.

> **Note**
>
> Since the IW41 confirmation only supports time-related units of measurement (for example hours and minutes), dummy units of measurement are displayed on the screen to make it more user friendly. However, the confirmation is posted to the back end with the Unit of Measure `Minutes` by default.

> **Note**
>
> Expense and Mileage entries are stored in the back end as confirmations. Due to limitations of the corresponding field in the back end, a single decimal space is supported for expense and mileage values.

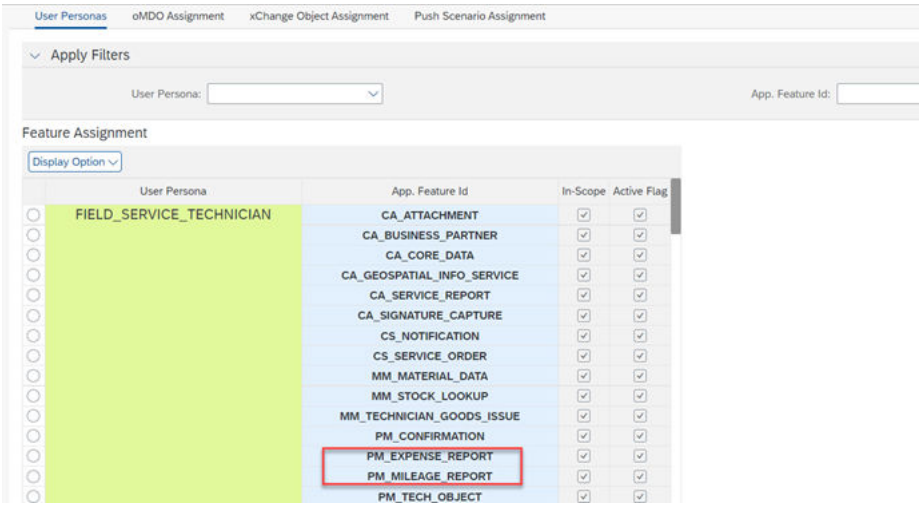
The parameters configured in the Configuring Mileage and Expense Parameters [page 308] procedure display as the *Activity Type* and *Work Center* fields in the app after you enter the order and operation number. If the *Work Center* parameter isn't defined, then this parameter is inherited from the operation. You can select a different *Activity Type* and *Work Center*, but it requires knowledge on the values that are selected to avoid errors when the data is posted to the back end.

8.2.1 Configuring Mileage and Expense Parameters

Prerequisites

Expense and mileage reporting is enabled on the mobile app when the following conditions are met:

- The user is assigned to the Field Service Technician persona.
- The *PM_EXPENSE_REPORT* and *PM_MILEAGE_REPORT* are enabled.



> **Note**
>
> You can enable the expense reporting and mileage reporting features for the Maintenance Technician persona.

Context

Use this procedure to configure the following parameters:

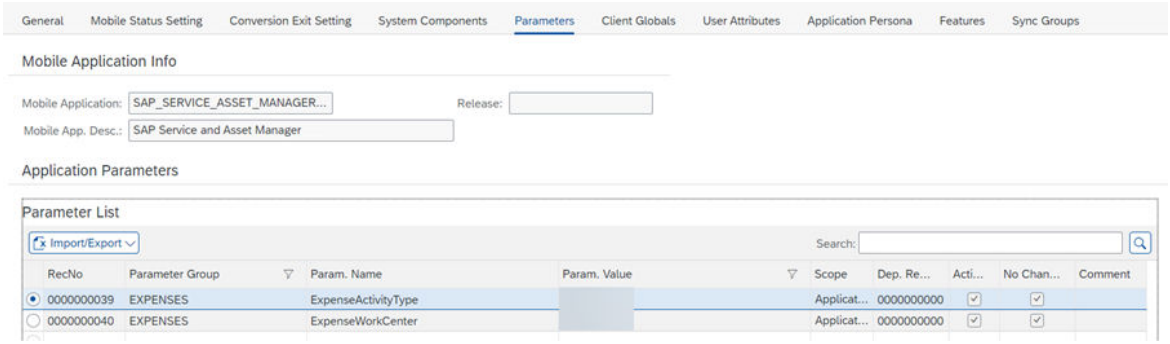
- *EXPENSES* group:
  - ExpenseActivityType: Activity type created for expenses. The defined value is the default displayed in the *Expense Type* field on the app.
  - ExpenseWorkCenter: The work center associated with the activity type defined for expenses.
- *MILEAGE* group:
  - MileageActivityType: Activity type created for mileage.
  - MileageWorkCenter: The work center associated with the activity type defined for mileage.
  - MileageUOM: The unit of measure (for example, miles or kilometers) displayed on the app.

Procedure

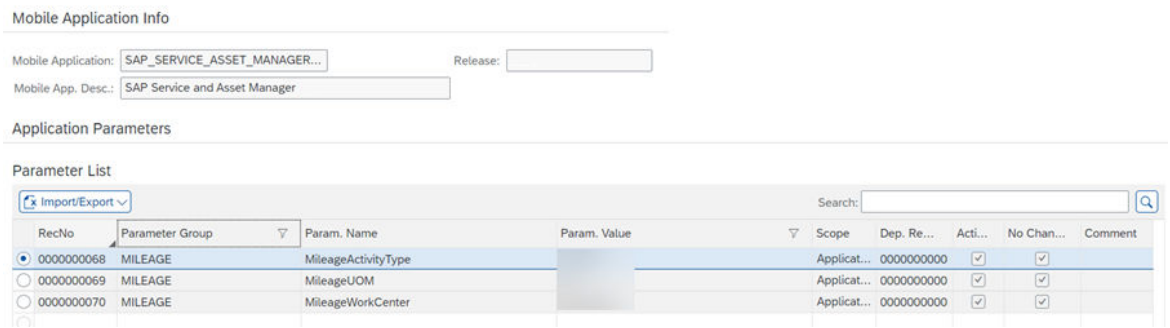
1. Using the ConfigPanel, navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.
2. The *ExpenseActivityType* and *ExpenseWorkCenter* parameters are found in the *EXPENSES* group. You can scroll down to find the parameter, or perform a search using the *Search* box. Highlight the desired parameter and click the *Change* button.
3. Configure the expense parameters as desired.

   > **Note**
   >
   > The UOM for expenses displayed on the SAP Service and Asset Manager screen is the currency field inherited from the order header.



4. The *MileageActivityType*, *MileageWorkCenter*, and *MileageUOM* parameters are found in the *MILEAGE* group. You can scroll down to find the parameter, or perform a search using the *Search* box.



5. Configure the mileage parameters as desired.
6. *Save* your changes.

8.2.2 Assignment Type for Planned Expenses and Mileage

When planned expenses and mileage are supported, certain assignment types are not impacted, while others must be addressed.

Since the planned expenses and mileage would involve adding additional operations, additional administrative steps or configuration may be needed to ensure that all operations, including expenses and mileage operations, are downloaded to SAP Service and Asset Manager.

Not Impacted Header Assignment Types

The following header assignment types are not impacted, because the entire order with all the operations is downloaded to the SAP Service and Asset Manager:

- Header Level Person Responsible
- Header Level Planner Group
- Header Level Business Partner
- Header Level Work Center

Assignment Types That Must Be Addressed

- **Operation Level Personnel Number**: To ensure that the planned expense and mileage operations are downloaded, ensure that each technician who works on the order is assigned to operations with expense and mileage activity types.
- **Sub-Operation Level Personnel Number**: To ensure that the planned expense and mileage operations are downloaded, ensure that each technician who works on the order is assigned to operations with expense and mileage activity types.
- **Capacity Requirements**: Planned operations with the expense and mileage activity types must be assigned to all technicians working on the order.
- **Operation Level Work Center**: If the customer has different work centers for mileage and expenses, a user attribute must be created to support multiple work center assignments. This user attribute must be assigned to the OPER_WORK_CENTER filter of the SAM2205_WORK_ORDER_GENERIC oMDO. Ensure that the expense and mileage work centers are also assigned to the technician via the User Administration section in the Administration and Monitoring Portal.

8.3 FSM Smartforms

SAP Service and Asset Manager supports FSM smartforms that allow planning board administrators to efficiently gather technician activity information in an organized way. Planning board administrators can assign

smartform instances to activities and require technicians to fill out the smartform instance before completing the activity.

Prerequisites

To use and preconfigure the FSM smartforms within SAP Service and Asset Manager, refer to the SAP Service and Asset Manager Integration with Field Service Management Scheduling [page 291] section.

8.3.1 Configuring FSM Smartform DTO Versions

MAIF has an oData Model that defines configurations and settings for sets of entities. Similarly, FSM has a data model for various entities, which are called DTOs. The DTOs have different versions as new features are developed for each DTO.

Context

Procedure

1. Log in to the *Administration and Monitoring Portal* in the backend (transaction code /SYCLO/ADMIN). On the *Administration* tab, select *Server Management*.
2. In the search parameters, enter:
   - Mobile Application = SAP Service and Asset Manager 2210
   - Server Name = SAM_FSM
3. Select the appropriate entry in the search results and navigate to the *Additional Properties* tab in the *Middleware Server Detail* section.
4. Add the following properties in the *Property List*:

Property Group	Property Name	Property Value
HEADERPARAM	ATTACHMENT_DTO_VERSION	14
HEADERPARAM	INSTANCE_DTO_VERSION	19
HEADERPARAM	ITEMPLATE_DTO_VERSION	19

The DTO versions can be changed to a newer version if necessary. For more information on DTO versions, refer to the https://help.sap.com/docs/SAP_FIELD_SERVICE_MANAGEMENT/fsm_data_model/data-model-overview.html documentation.

5. Save your changes.

8.3.2 Smartform Instance and Template oMDO filters

Context

Procedure

1. Log in to the Configuration Panel in the backend (transaction code /SYCLO/CONFIGPANEL).
2. Navigate to *oData Mobile Data Object Configuration* > *SAP_SERVICE_ASSET_MANAGER_2210* > *SAM2210_FSM_FORM_INSTANCE*.
3. Navigate to the *Data Filter tab* > *Operation – READ* > *Standard Filter*.
4. The following filters should be noted:
   - MAX_INSTANCES: This filter controls the maximum number of instances that can be loaded during synchronization.
   - GET_CLOSED: If this filter is set to true, closed smartform instances will also be loaded.
5. To configure MAX_TEMPLATES, which has the same functionality as MAX_INSTANCES but for smartform instances, follow all previous steps but search for the SAM2210_FSM_FORM_TEMPLATE oMDO.

9 Field Service Technician Configuration on an S/4 Field Service Management Back End

9.1 Configuring Connectivity between SSAM and FSM

Prerequisites

Ensure that a basic integration structure exists between the S/4 Service module in the S/4HANA backend and the FSM to enable data replication between the SAP backend and the FSM.

The following replication scenarios must already be active and implemented between the SAP ECC or S/4HANA backend and the FSM:

1. Status Updates: Status updates from SSAM will be transmitted to the FSM Planning and Dispatching Tooll. Since the FSM only supports Activity (Item) level assignment, status transitions are only supported for assigning a service item to a business partner. Only statuses mapped to FSM service workflow statuses will be transferred to the FSM.

2. Technician Location Tracking: The technician location obtained from SSAM will be updated in FSM so that it can be seen on the FSM Service Map.

3. FSM Smart Forms: FSM Smart Forms can be assigned to a service item, downloaded and completed in SSAM.

> **Note**
>
> Rejection and transfer of items is not supported in SSAM with FSM scheduling integration scenario.

For more information, refer to the Integration with SAP Field Service Management topics.

Context

Procedure

1. Create a client in SAP FSM

- Log on to your FSM administrative URL
- Create a new client with the following parameters:
  - Client Authentication Method = CLIENT_SECRET
  - User Groups = Admin
  - Save the Client Id and Client Secret

2. Configure STRUST in the SAP Backend
   - Download the ISRG Root X1 certificate. The .pem file can be downloaded.

   > **Note**
   >
   > For more information, refer to the Announcing Root Certificate Change topic.

   - Install the ISRG Root X1 certificate against SSL Client (Anonymous) in the Trust Manager
3. Create RFC Destinations in the SAP Backend
   1. Create a destination for establishing the oAuth connection between the SAP backend and FSM with the following parameters:
      - Name: <e.g. SSAM2310_FSM_OAUTH>
      - Connection Type: G (HTTP Connection to External Server)
        Under Technical Settings:
        - Target Host: auth.coresuite.com
        - Path Prefix: /api/oauth2/v1/token

        Under Logon & Security:
        - Select Basic Authentication
        - In the User and Password section, enter the client ID and client secret from step 1
        - In the Security Options section, specify the following:
          - SSL = Active
          - SSL Certificate = ANONYM SSL Client (Anonymous)
   2. Create a destination for implementing the API calls to FSM with the following parameters:

   - Name: <e.g. SSAM2310_FSM_API>
   - Connection Type: G (HTTP Connection to External Server)
     Under Technical Settings:
     - Target Host: us.coresuite.com

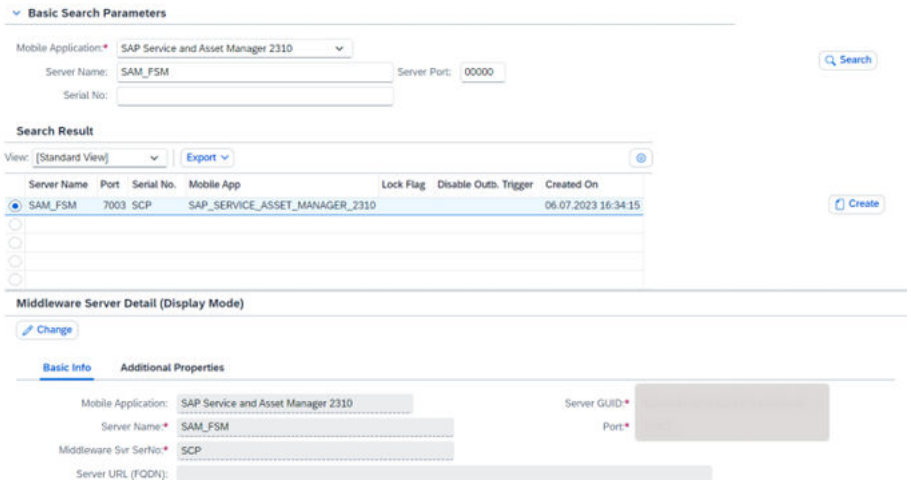
     Under Logon & Security:
     - Select Do not use a user
     - In the Security Options section, specify the following:
       - SSL = Active
       - SSL Certificate = ANONYM SSL Client (Anonymous)

9.2 Configuring Server Parameters

Context

Procedure

1. Log in to the administration and monitoring portal in the backend (transaction /SYCLO/ADMIN). On the *Administration* tab, select *Server Management*.
2. Create a new server with the name SAM_FSM for the application SAP_SERVICE_ASSET_MANAGER_2310.



3. On *Server Properties* tab, save the following properties:

Property Group	Property Name	Property Value
AUTHENTICATION	RFC_OAUTH_S4	<Name of the RFC in Step 3.a> RFC_OAUTH_S4: <e.g., SSAM2310_FSM_OAUTH>
AUTHENTICATION	RFC_API_S4	<Name of the RFC in Step 3.b> RFC_API_S4: <e.g., SSAM2310_FSM_API>
HEADERPARAM	ACCOUNT_S4	<FSM Account Name>
HEADERPARAM	ACCOUNT_ID_S4	<FSM Account Id>
HEADERPARAM	COMPANY_S4	<FSM Company Name>
HEADERPARAM	COMPANY_ID_S4	<FSM Company ID>

Property Group	Property Name	Property Value
HEADERPARAM	CLIENT_ID	<Client ID to send to FSM. E.g SY-Sid.Client>
HEADERPARAM	CLIENT_VERSION	1.0

The following additional server parameters can be defined as needed, but they are not mandatory. Their default values are listed below. Changing these values may result in having to refine the backend integration code.

Property Group	Property Name	Property Value	Description
AUTHENTICATION	AUTH_TYPE	OAUTH	Authentication type used to communicate with FSM – Default OAUTH
DTO	STATUSDEF	ServiceAssignmentStatus-Definition.9	FSM Data Transfer Object (DTO) version. Can be changed if a newer version of a DTO is needed.
	ITEM	Item.23	
	EQUIPMENT	Equipment.23	
	BUSINESSPARTNER	BusinessPartner.23	
	ADDRESS	Address.21	
	PERSON	Person.24	
	SERVICEASSIGN	ServiceAssignment.28	
	SERVICECALL	ServiceCall.26	
	ACTIVITY	Activity.39	
	SERVICEASSIGNSTATUS	ServiceAssignmentStatus.15	
INTEGRATION	CREATEPERSON	admin	User to create new objects in FSM (applies to SSAM only)
UI_URL_PATH	ADDRESS	/api/data/v4/Address	URI strings for calling FSM api
	EQUIPMENT	/api/data/v4/Equipment	
	ITEM	/api/data/v4/Item	
	PERSON	/api/data/v4/Person	
	BUSINESSPARTNER	/api/data/v4/Business-Partner	
	SERVICEASSIGN	/api/data/v4/ServiceAssignment	
	QUERY	/api/query/v1	
	SERVICEASSIGNSTATUS	/api/data/v4/ServiceAssignmentStatus	
	SERVICECALL	/api/data/v4/ServiceCall	

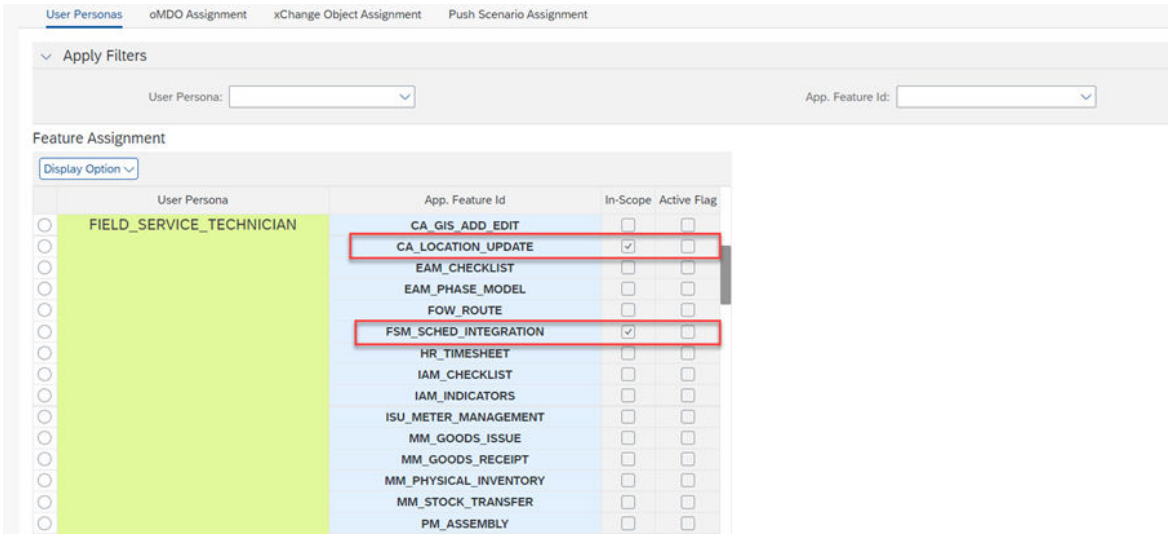
Property Group	Property Name	Property Value	Description
	SERVICEORDERCREATE	/service-management/api/v2/composite-tree/service-calls?autoCreateActivity=false	
	ACTIVITY	/api/data/v4/Activity	

9.3 Configuring Persona

Context

The following configurations are required to use the FSM scheduling integration scenarios in FSM:

1. You must add the FIELD_SERVICE_TECHNICIAN persona assigned to the user.
2. In the configuration panel, navigate to the component assignment for the mobile application and ensure that the FSM_SCHED_INTEGRATION and CA_LOCATION_UPDATE features are enabled, as they are disabled by default.



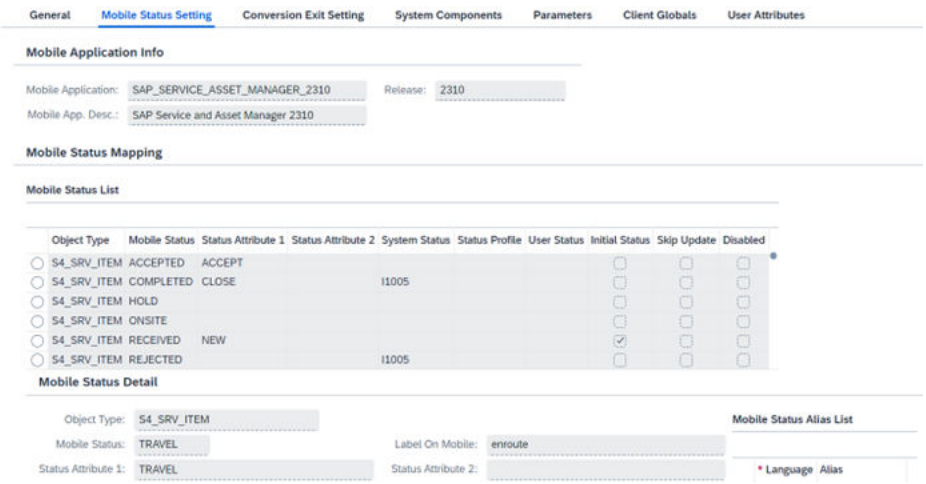
9.4 Configuring Statuses

Context

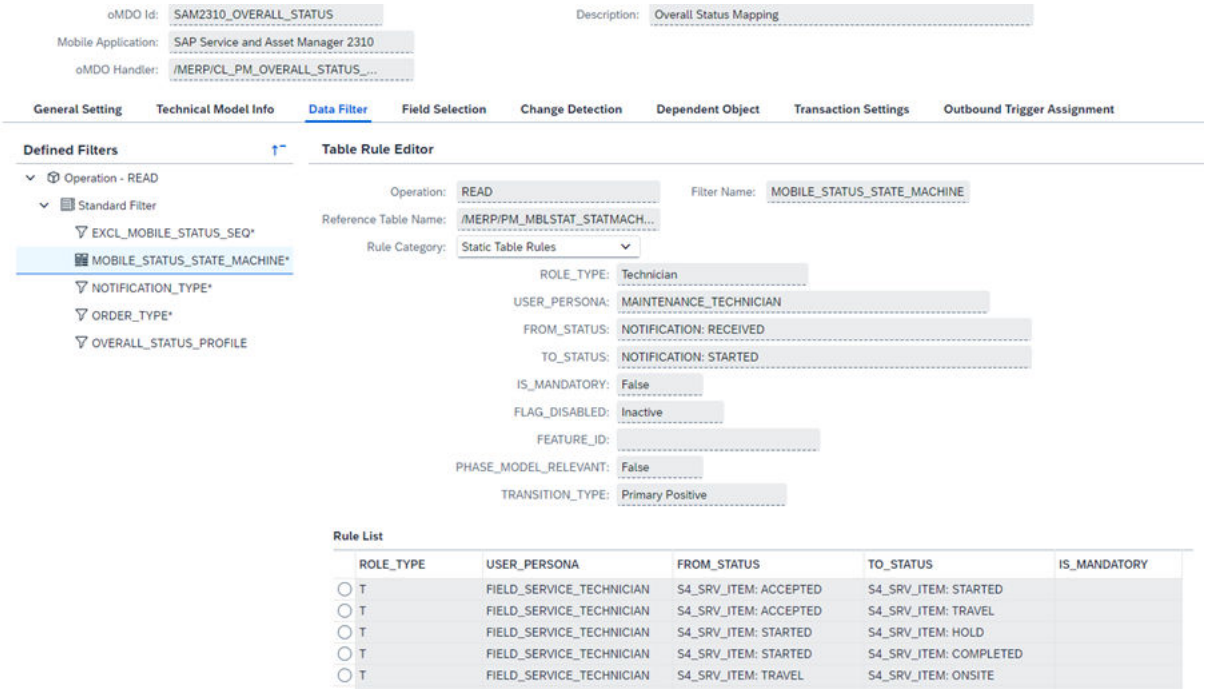
For the FSM scenario, only assignment type 2 is supported, and status changes are therefore only supported at the operation level. The default SSAM statuses supported for the Field Service Technician persona are defined in the *Mobile Application* > *Mobile Status Settings* configuration:

Object Type	Mobile Status	Initial Status (Y/N)	Status Attribute 1 (FSM status)
S4_SRV_ITEM	RECEIVED	Y	new
S4_SRV_ITEM	ACCEPTED		accept
S4_SRV_ITEM	TRAVEL		travel
S4_SRV_ITEM	ONSITE		
S4_SRV_ITEM	STARTED		work
S4_SRV_ITEM	HOLD		
S4_SRV_ITEM	COMPLETED		close
S4_SRV_ITEM	TRANSFER	No status update	

If any additional SSAM statuses are used or a custom service workflow is applied in the FSM then additional statuses can be configured in the *Mobile Application* > *Mobile Status Settings* configuration. In this case it may require customization at the back end and/or MDK level depending on how the statuses are to be used.



The default status transitions are defined in the oMDO SSAM2310_OVERALL_STATUS, filter MOBILE_STATUS_STATE_MACHINE. If additional status transitions are required or existing ones need to be modified, the filter values in this oMDO must be changed. This may require customization at the back end and/or MDK level depending on how the status transitions work.

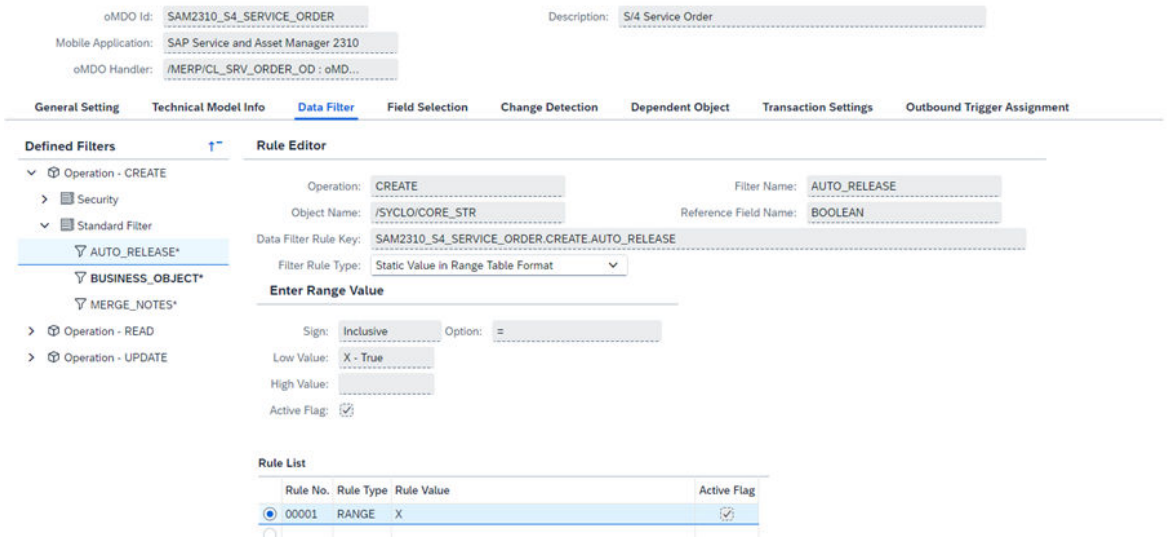


9.5 Auto Release of S4 Service Orders

When a service order is created in SSAM, you can configure that the service order is automatically released when it is published in SAP.

To set up automatic release of an order, you must perform the following configuration:

- Configure the AUTO_RELEASE filter for the CREATE operation in oMDO SAM2310_S4_SERVICE_ORDER - Filter I = 'X' must be active.



10 Linear Asset Management Configuration

Linear assets are technical systems with a linear infrastructure whose condition and properties can vary from section to section (dynamic segmentation). You can see linear asset data associated with various objects such as work orders, operations, technical objects, and notifications.

In addition to the basic SAP Service and Asset Manager configuration, there are a few considerations when configuring SAP Service and Asset Manager with Linear Asset Management (LAM).

Configuration items include:

- Disabling and re-enabling LAM
- Enabling technical objects work list feature
- Adding the LAM equipment type filter
- Adding linear data for statistics

Each of these configuration items is described in the sections that follow.

> **Note**
>
> Before configuring LAM on your system, see 2900476 so that the LAM exchange works properly.

10.1 Disabling and Re-Enabling Linear Asset Management

Linear Asset Management (LAM) is especially designed to meet the requirements of linear asset maintenance. A linear asset is a special type of asset that has an associated length dimension. This dimension is represented through starting and ending points or by specifying the asset length. For the mobile device, the linear asset management functionality enables the field technician to work on orders and notifications that have linear equipment and functional locations. Field personnel can create work orders, notifications, time confirmations, and material confirmations for the linear assets.

From the mobile device, you can view linear data for the following:

- Work orders
- Operations
- Notifications
- Items
- Equipment
- Functional locations
- Confirmations
- Measuring points
- Measurement documents

You can create or edit linear data on characteristic values.

The Linear Asset Management feature is called *PM_LINEAR_ASSET_MANAGEMENT*. By default, the feature is not enabled.

> **Note**
>
> As of the SAP Service and Asset Manager 2205 release, enable and disable parameters are no longer available through the *Parameters* tab. You enable or disable all features through the *Features* tab. See the Configuring Features [page 90] procedure for details.

Use the following OMDOs to configure your LAM feature:

- LAM_OBJECT_DATA
- LAM_OFFSET_TYPE
- LINEAR_REFERENCE_PATTERN

For general information on configuring OMDOS, see the *OData Channel Integration Settings Procedures* topics found in the *Mobile Add-On Configuration Configuration Panel Common Procedures* chapter of this guide.

10.2 Adding Linear Data for Characteristics

The functionality of adding or editing linear data for characteristics defines segments of a linear asset where a specific attribute, or characteristic value, is valid. A segment is defined by start point, end point, length, and unit of measurement (linear data).

Relevant OMDO: **SAM2310_LAM_CHAR_VALUE**.

> **Note**
>
> - Linear data for characteristics works only for characteristics that are marked as relevant for linear asset management.
> - To use linear data for characteristics, create a special *Organizational Area* and assign it to the relevant classes and characteristics.
> - It's possible to assign several characteristic values in different segments of a linear asset. Therefore, set the value assignment to *multiple values* when the characteristics are created.

11 Meter Management Configuration

The Meter Management component is delivered out of the box with predefined settings, which you can change according to your back-end system setup. The following settings, however, have to be set:

- Binding Industry Solutions & Utilities (ISU) process type to work order type
- Setting the optimal meter reading history
- Binding meter reading reason relevant for technical installation
- Binding meter reading notes based on the ISU process type

11.1 Binding ISU Process Type to Work Order Type

Binding the ISU process type to the work order type provides the SAP Service and Asset Manager application the correct representation of what process type is being conducted with the different work order types. This binding is located in the **SAM2310_ORDER_ISULINK** OMDO, under the *Read* filter. You can update these filters according to your business process.

All assignment types are supported for the Meter Management component.

The following filters represent binding criteria for different process types. By default, the ISU process types are bound to order types as follows:

Filter Name	Type	Value	Comments
ORDTYPE_DISCONNECT	Standard Filter Mandatory	DC01 RC01	Used for the disconnect process
ORDTYPE_INSTALL	Standard Filter Mandatory	SM01	Used for the installation process
ORDTYPE_READING	Standard Filter Mandatory	MR01	Used for the meter reading process
ORDTYPE_REMOVE	Standard Filter Mandatory	CU01	Used for the remove process
ORDTYPE_REPAIR	Standard Filter Mandatory	SM02	Used for the repair process

Filter Name	Type	Value	Comments
ORDTYPE_REPLACE	Standard Filter Mandatory	RP01	Used for the replace process

To change the default binding for a particular process, complete the steps below:

1. On the ConfigPanel home page, choose *OData Mobile Data Object Configuration*.
   Make sure that you select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. From the *OData Mobile Data Object List* select desired OMDO object, such as **SAM2310_ORDER_ISULINK**, and then click the *Data Filter* tab.
3. Expand the *Defined Filters* list under the *READ* operation with the standard filter. Select the filter that you want to update from the list of available filters as listed in the table in this topic. Choose the *Change* button from the menu.
4. Set the order type for the desired process type you have selected.
5. Save your changes.

11.2 Binding Meter Reading Notes Based on ISU Process Type

The back end ISU system configuration specifies which meter reading notes are relevant for the major ISU process types. This configuration has to be replicated in the OData Mobile Data Object **SAM2310_METER_READING_NOTE** under the *READ* operation with the standard filters, so that the SAP Service and Asset Manager application reflects the proper meter reading notes for a specific process type. The default configuration lists the meter reading notes relevant for the ISU process. However, you can change it if you have different requirements in your back end configuration.

Filter Name	Type	Value	Comments
DISCONNECT_NOTE_REMOVE	Standard Filter, Mandatory	01 04 05	Used for the disconnect notes for the remove process
METERREAD_NOTE _INSTALL	Standard Filter, Mandatory	01 04	Used for the meter reading notes for the installation process
METERREAD_NOTE _REMOVE	Standard Filter, Mandatory	04 05	Used for the meter reading notes for the remove process

To change the default binding for a particular process, complete the following steps:

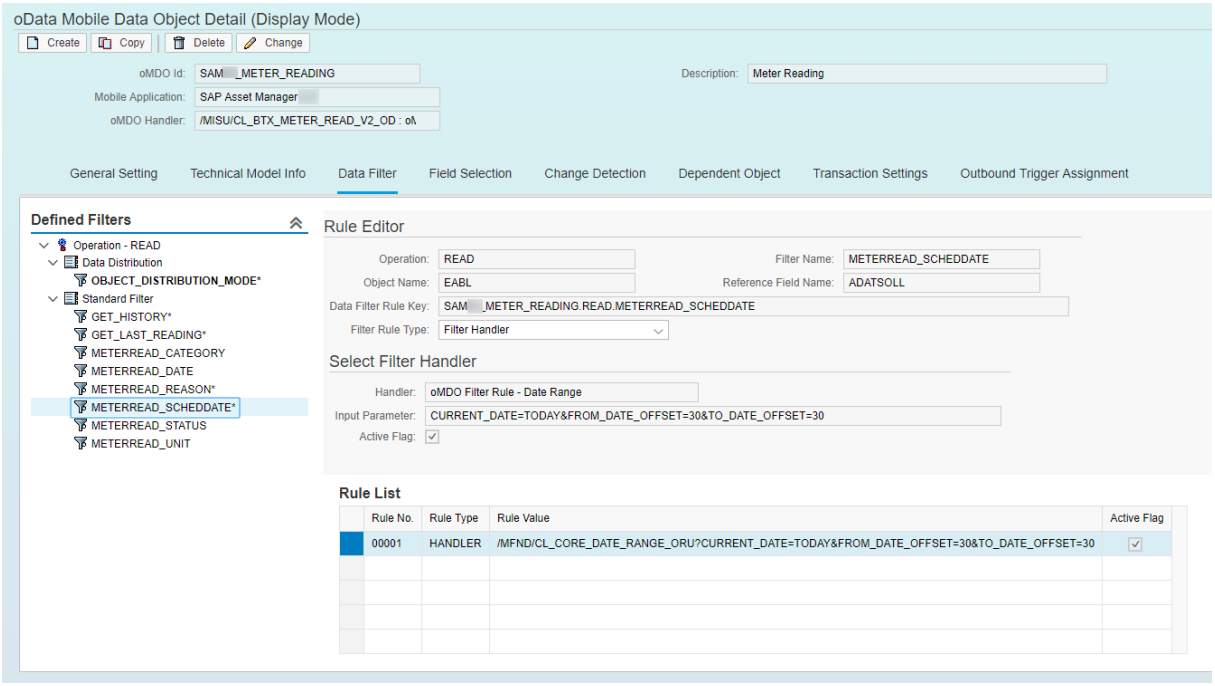
1. On the ConfigPanel home page, choose *OData Mobile Data Object Configuration*.

Make sure you select your desired mobile application in the *Mobile Application Filter* field at the top of the page.

2. From the *OData Mobile Data Object List* select desired OMDO object, such as **SAM2310_METER_READING_NOTE**, and then click on the *Data Filter* tab.
3. Expand the *Defined Filters* list under the *READ* operation with the standard filter. Select the filter that you want to update from the list of available filters as listed above. Choose the *Change* button from the menu.
4. Set the meter reading note for the desired ISU process type you have selected.
5. *Save* your changes.

11.3 Meter Reading History Interval Definition

The default setting for meter reading history is to include all meter readings from the past 30 days till the current day. If you have a different requirement, you can change it from the OData Mobile Data Object **SAM2310_METER_READING** under the READ operation with the standard filter *METERREAD_SCHEDDATE* as shown in the following example.

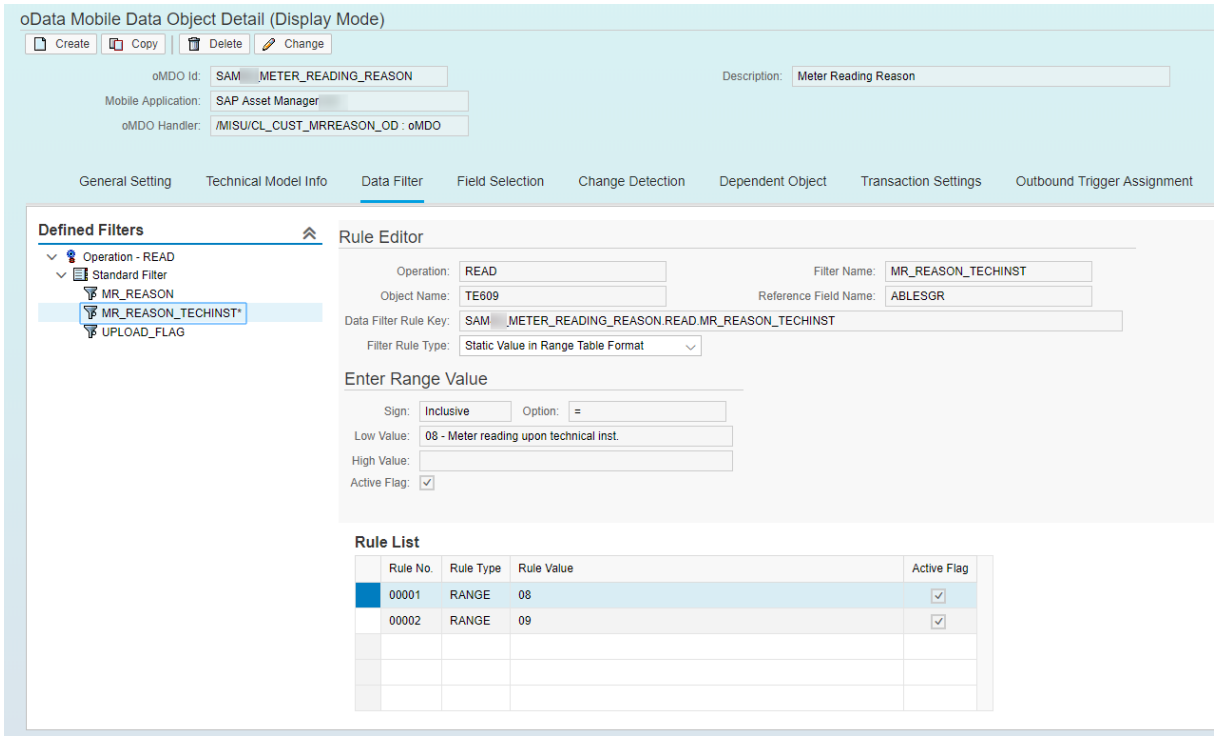


11.4 Binding Meter Reading Reasons Relevant for Technical Installation

The back end ISU system configuration specifies explicitly which meter reading reasons are relevant for technical installation. These are the only reasons displayed on the SAP Service and Asset Manager application when completing a meter reading during the technical installation process. Set this binding in the OData

Mobile Data Object **SAM2310_METER_READING_REASON** under the *READ* operation with the standard filter *MR_REASON_TECHINST*.

The default configuration contains meter reading reasons *08* and *09* as required for technical installation. If you have different requirements for your back end configuration, you can change these defaults.



12 Quality Management Configuration

The Quality Management application component supports tasks associated with quality planning, quality inspection, and quality control. In addition, it controls the creation of quality certificates and manages problems with the help of quality notifications.

Quality Management (QM) notifications are integrated with SAP Service and Asset Manager as follows:

- Display QM type notifications on the mobile device
- Create QM notifications at the inspection lot level
- Create QM notification items to record defects noted in inspections
- Maintain the association of notifications and notification items with the characteristics

> **Note**
>
> Before configuring QM, you must install the QM component. See the Quality Management chapter in the *SAP Asset Manager Component Installation Guide* for the installation procedure.

Quality Management is disabled on an out-of-the-box new SAP Service and Asset Manager installation. Use the following topics and procedures found in this chapter to enable Quality Management.

12.1 Configuring Quality Management Notification Parameters

Context

Code groups that belong together in terms of content are grouped in catalogs. These catalogs are identified by the catalog type (a number or a letter). For example, in this way, you combine:

Use the following parameter groups and associated parameters to configure Quality Management in SAP Service and Asset Manager:

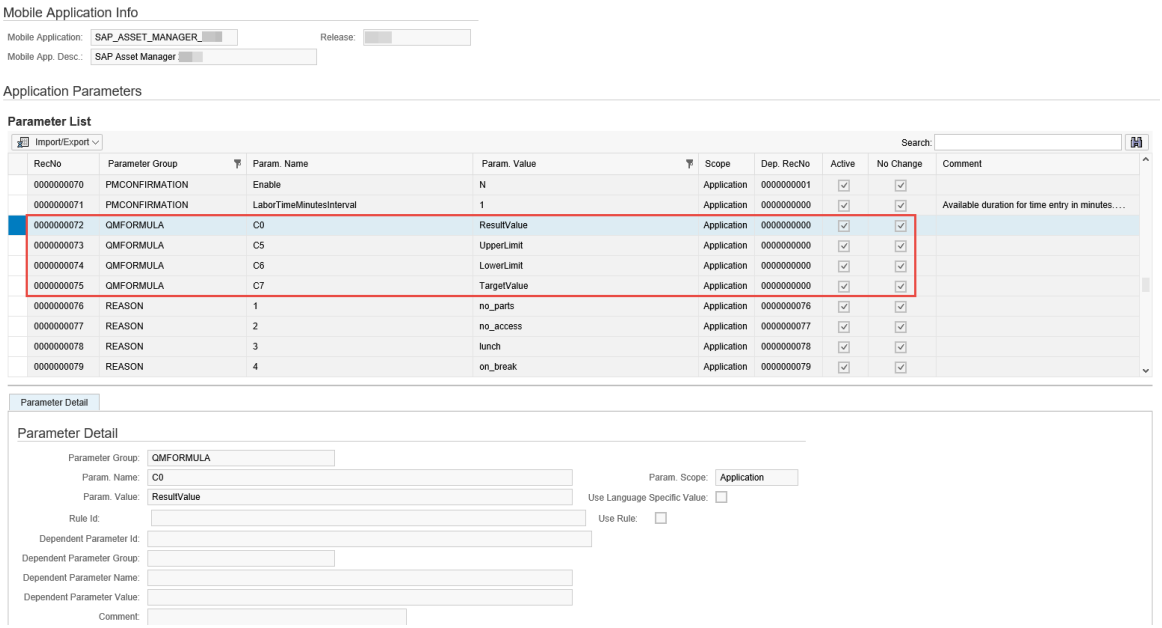
- **QMFORMULA:** Inspection lots can contain points or characteristics. You can use the following formula parameters to determine the characteristic value:
  - **C0:** Arithmetic mean of measured values for characteristic. Standard parameter.
  - **C5:** Upper limit of tolerance range
  - **C6:** Lower limit of tolerance range
  - **C7:** Target value of characteristic
- **CATALOGTYPE:** Sets a usage decision on inspection lots on the client. Standard is *3*
- **DOCUMENT:** Standard is *QMQMEL*

- **INSPECTION_LOT:** Controls if the user of the application can set a decision to complete an inspection lot. When you set an inspection lot to complete, defects are also automatically completed. Standard is *N*.

Procedure

1. Using the ConfigPanel, navigate to *Mobile Application Configuration* > *Parameters tab*. In the left column, *Defined Mobile Applications*, select your application.

   The *Parameter List* populates with a list of all parameters available for the application.
2. Type your desired parameter group into the *Search* box, or scroll to find your parameter. Highlight the parameter you want to configure and click the *Change* button. The following example uses the parameter group *QMFORMULA*.



3. Make your desired parameter association changes.
4. Check the `<Active>` flag to ensure that the parameter is used by the mobile application. If desired, and if not already checked, check the `<No Runtime Change>` box to ensure that the value of the parameter isn't overridden at runtime through synchronization processing.
5. *Save* your changes.

Results

Your Quality Management parameters are configured and enabled.

Next Steps

Complete all other topics and procedures in the Quality Management Configuration [page 326] to fully configure and enable the Quality Management feature.

12.2 Configuring oData MDO Data Filters for Quality Management

For general information and a procedure on changing OMDO data filters, see the following topics:

- Working with oData MDO Filter Rules [page 149]
- Changing oData MDO Filter Rules [page 149]

To fully enable Quality Management, ensure that the following data filters and data filter rules are configured properly and active:

SAM2310_CATALOG_CODES

Select **SAM2310_CATALOG_CODES** from the *oData Mobile Data Object List* and navigate to ▶ *Operation - READ* ▶ *Standard Filter* ▶ *CATALOG_TYPE* ▶. Ensure the following rules in the `<Rule Value>` field are set to *Active*:

- 8 - Activities (QM)
- 9 - Defect Types
- D - Coding

SAM2310_DOCUMENT

When you enable QM on the **SAM2310_DOCUMENT** data filter, you enable the ability to attach documents to an inspection. When mobile client users take a reading on the SAP Service and Asset Manager application, they can attach documents that relate to the inspection on work orders, equipment, functional locations, or notifications.

Select **SAM2310_DOCUMENT** from the *oData Mobile Data Object List* and navigate to the following locations:

- ▶ *Operation - CREATE_MEDIA* ▶ *Data Segment* ▶ *DOCUMENT_SWITCH* ▶
- ▶ *Operation - READ* ▶ *Data Segment* ▶ *DOCUMENT_SWITCH* ▶

Set the following objects in the *Rule List* to *Active*:

- QMQEL: QM notification
- QMTBLOC: Inspection method

Ensure that both *BDS* and *GOS* are set to *Active*.

Navigate to *Operation - READ* > *Standard Filter* > *DOC_LINK_OBJ* > and ensure the following are set to *Active*:

- QMTBDOC: Inspection method attachment
- QMQMEL: QM notification attachment

SAM2310_INSPECTION_LOT

Set the dependency to get inspection lots based on work orders related to the inspection assigned to the mobile client user as follows:

Select **SAM2310_INSPECTION_LOT** from the *oData Mobile Data Object List* and navigate to *Operation - READ* > *Data Distribution* > *OBJECT_DISTRIBUTION_MODE* . Ensure that `<Range Value>` *2 - Dependency Queue* is set to *Active*.

Navigate to the *Dependent Object* tab. Ensure that `<Source Tech Entity Type>` *INSPECTIONLOT* is connected to `<Dependent oMDO ID>` **SAM2310_NOTIFICATION_GENERIC** and is set to *Active*. When a mobile client user fetches the inspection lots, the related defects, created as QM notifications, are fetched as well.

Ensure that `<Source Tech Entity Type>` *INSPECTIONCHAR* is connected to `<Dependent oMDO ID>` **SAM2310_INSPECTION_HISTORY** and is set to *Active*. The fetched inspection history is based on the inspection characteristics available.

SAM2310_NOTIFICATION_GENERIC

Select **SAM2310_NOTIFICATION_GENERIC** from the *oData Mobile Data Object List* and navigate to *Operation - READ* > *Data Distribution* > *NOTIF_ASSIGNMENT_TYPE* . Ensure that `<Range Value>` *D* is active.

Navigate to *Operation - READ* > *Data Distribution* > *NOTIF_TYPE* . Ensure that `<Input Parameter>` *QM* is active.

SAM2310_NOTIF_PARTNER_DET_PROC

Enabling QM in the **SAM2310_NOTIF_PARTNER_DET_PROC** object means that selecting a business partner is mandatory when a user creates a QM notification.

Select **SAM2310_NOTIF_PARTNER_DET_PROC** from the *oData Mobile Data Object List* and navigate to *Operation - READ* > *Standard Filter* > *NOTIF_TYPE* . Ensure that `<Input Parameter>` *QM* is active.

SAM2310_NOTIF_TYPE

Select **SAM2310_NOTIF_TYPE** from the *oData Mobile Data Object List* and navigate to *Operation - READ* > *Standard Filter*. Ensure that Quality Management is enabled in the following locations:

- **NOTIF_CATEGORY:** Ensure that `<Range Value>` *02 - Quality Notification* is active.
- **NOTIF_PRIORITY_TYPE:** Ensure that `<Range Value>` *QM - QM Priorities* is active.
- **NOTIF_TYPE:** Ensure that `<Input Parameter>` *QM* is active.

SAM2310_PRIORITY

Select **SAM2310_PRIORITY** from the *oData Mobile Data Object List* and navigate to *Operation - READ* > *Standard Filter* > *PRIORITY_TYPE*. Ensure that `<Rule Value>` *QM* is active.

SAM2310_WORK_ORDER_GENERIC

Enable the dependency between a work order and a related inspection lot on the mobile device:

Navigate to the *Dependent Object* tab. Find the *WOHEADER* objects in the `<Source Tech. Entity Type>` column. Scroll to find the **SAM2310_INSPECTION_LOT** object. Ensure it is marked as *Active*.

13 SAP Business Application Studio (instead of SAP Web IDE)

SAP Business Application Studio is the successor product of SAP Web IDE and includes all the SAP Web IDE capabilities (The UI might be different), as well as new and improved ones.

> **Note**
>
> SAP Web IDE is replaced by SAP Business Application Studio. For general information, refer to SAP Web IDE Sunset tutorial guide.

SAP Business Application Studio is a powerful cloud-based Integrated Development Environment (IDE) provided by SAP. It empowers developers to easily build, test, and deploy enterprise-grade applications on the SAP Business Technology Platform (BTP). It is particularly well-suited for developing with the SAP Cloud Application Programming Model, SAP Mobile Services, and SAP frontend technologies like SAP Fiori elements, SAPUI5 and the SAP Mobile Development Kit. With a comprehensive suite of development tools, frameworks, and services, developers can streamline their development process and accelerate time-to-market. A full suite of development tools – including code editors and version control – all accessible through a web browser.

SAP Business Application Studio clearly stands out for its tight integration with SAP services, technologies, and systems. Central to the architecture of the development environment are the Dev Spaces. These are specialized IDEs in the cloud that come pre-installed with everything you need for specific use-cases in SAP environments like building SAP Fiori extensions or HANA native development. SAP Business Application Studio provides clear guidance following SAP best practices and eases developers into the world of SAP technologies.

> **Note**
>
> To get started with SAP Business Application Studio, refer to the SAP Business Application Studio help portal.

13.1 Migrating to SAP Business Application Studio

SAP provides support for customers migrating from the Neo environment to the multi-cloud environment.

> **Note**
>
> For general information about migration, see Migrating from the NEO Environment to the Multi-Cloud Foundation. In addition, there are specific SAP Web IDE migration guides covering different use cases.
>
> See the individual use case migration guide for details, Migrating Your Mobile Development Kit (MDK) Apps from SAP Web IDE to SAP Business Aplication Studio.

SAP provides guides, documentation, and migration wizards to facilitate the migration process from SAP Web IDE to SAP Business Application Studio, but in some cases, manual migration is required. The effort to migrate

your applications varies depending on the number of applications, complexity, use case, and internal company processes.

13.2 Customizing Apps Using the Mobile Development Kit

The Mobile Development Kit for SAP Business Technology Platform Mobile Services is a metadata-based application development platform.

The Mobile Development Kit (MDK) lets you customize, deploy, and manage your customized Windows apps in the cloud. The Mobile Development Kit editor lets you edit various aspects of your application using the Mobile Development Kit editor. It also provides native client support and consumes mobile services such as onboarding, offline OData, life-cycle management, and supportability through the SAP Business Technology Platform Mobile Services using the Mobile Development Kit client.

The Mobile Development Kit allows business process experts to customize the app in a cloud-based editor using the SAP Business Application Studio, and developers to edit code directly in the metadata files.

The end-to-end use case for Mobile Development Kit includes tasks involving the following roles:

- Administrator
- Business process expert
- Developer
- User

13.3 Customizing Metadata Using the Mobile Development Kit

One of the main purposes of the Mobile Development Kit is to easily customize and redeploy metadata for your SAP Service and Asset Manager application.

> ⚠ Restriction
>
> Develop any customization on the app as a separate component in a Mobile Development Kit project. Developing customizations as a component makes it easier to maintain customizations during upgrades, as it isolates custom code. Isolating your custom code eliminates the chance of overwriting when you implement a new release.

A typical metadata customization procedure is as follows. This example assumes that metadata definitions already exist in the Mobile Development Kit and that you're customizing them, or changing them:

1. Locate the object you want to modify. You can modify pages, actions, or rules. See the following topics and subtopics for more information on how to create and modify the following metadata objects using the Mobile Development Kit:
   - Create Pages
   - Create Actions

- Create Rules

2. Deploy the metadata. See Creating Apps in SAP Business Application Studio (Cloud Foundry) for more information.

13.3.1 Customizing Contextual Menu Options

You can configure the context menu using rules in the metadata.

Use the following rules in the metadata to customize the contextual menu options:

- `Rules/ContextMenuTrailingItems.js`
- `Rules/ContextMenuLeadingItems.js`
- `Rules/ContextMenuTable.js`

The following contetual menu items are preset in the app:

Object	Assignment Type	Leading - Item 1	Leading - Item 2	Trailing - Item 1 (Synced / Local)	Trailing - Item 2 (Synced / Local)
Work Order	1, 5, 7, 8		Transfer / Hold	Edit	- / Delete
	2, 3, 4, 6	Take Reading		Edit	- / Delete
Operation	2, 4, 6		Transfer / Hold	Edit	- / Delete
	1, 3, 5, 7, 8	Confirm / Uncon-firm		Edit	- / Delete
Suboperation	3		Transfer / Hold	Edit	- / Delete
	1, 2, 4, 5, 6, 7, 8	Confirm / Uncon-firm		Edit	- / Delete
Notification				Edit	- / Delete
Equipment		Add Notification	Add Work Order	Take Readings	
Functional Loca-tion		Add Notification	Add Work Order	Take Readings	
Documents				Delete (Local)	
Reminders				Delete	
Errors				Delete	

For detailed information on working with metadata contained in the Mobile Development Kit, see the SAP Mobile Development Kit API References portal, specifically, the ObjectCell control

13.3.2 Enabling Multi-User Support

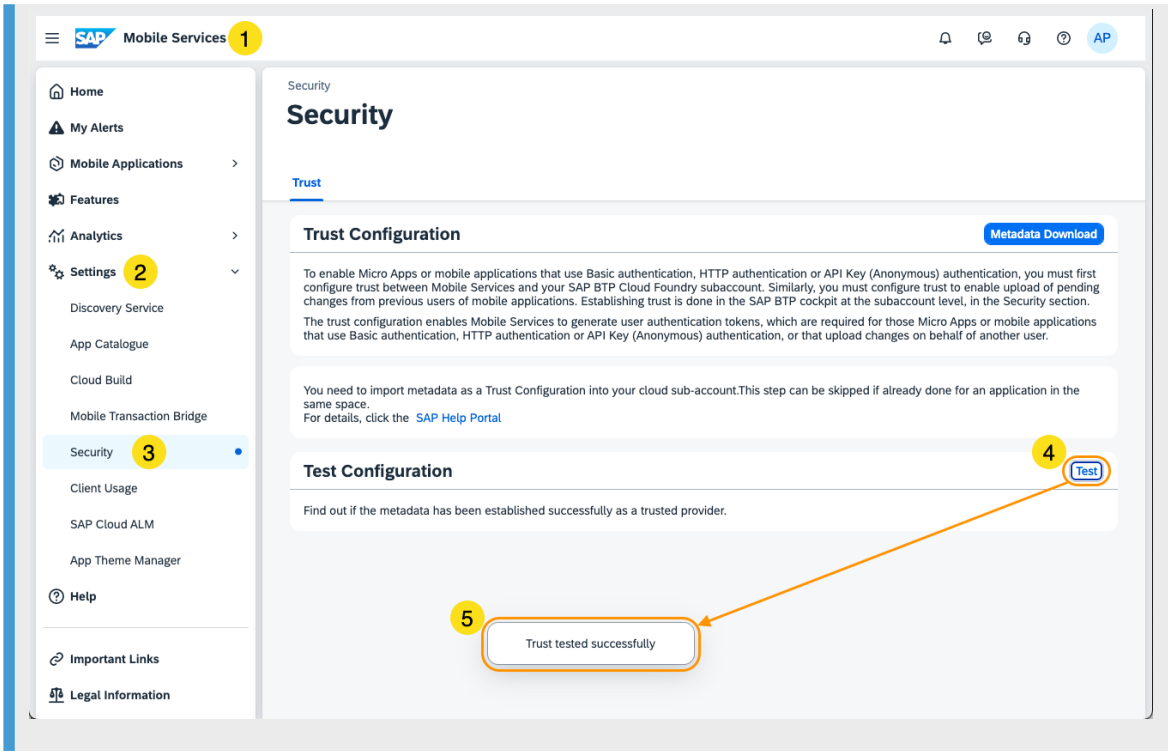
Multiple users can use the client on the same device, capabilities such as adding a new user, switching across users, and searching from a list users are supported.

Procedure

1. Set *MultiUserEnabled* to *True* in *MDKClient_SDK/branding/BrandedSettings.json* > *ConnectionSettings* > *MultiUserSettings*.
   "ConnectionSettings": { "AppId": "ODataOnly", ... "MultiUserSettings": {"MultiUserEnabled": true
2. Set *Allow Upload of Pending Changes from Previous User* for the corresponding app to *True* in the mobile services cockpit.

   > **Note**
   >
   > *Allow Upload of Pending Changes from Previous User* ensures that any pending offline changes from the previous user are securely uploaded to the service back end. This is important when the previous user has not uploaded the offline changes, and the app is switched to a new user. Also, you must apply the following, depending on which environment you use:
   >
   > - **Cloud Foundry**:
   >   - *Allow Upload of Pending Changes from Previous User* is available in *Mobile Settings Exchange* feature under *Shared Devices* section.
   >   - When you enable *Allow Upload of Pending Changes from Previous User*, you must also enable *Passcode Policy*. *Passcode Policy* checkbox is under *Assigned Features* > *Mobile Settings Exchange*.
   > - **Neo**:
   >   - *Allow Upload of Pending Changes from Previous User* is available in *Client Policies* feature under *Shared Devices* section.
   >   - When you enable *MultiUserEnabled*, you must also enable *Allow Upload of Pending Changes from Previous User* and *Passcode Policy* to be able to onboard. *Passcode Policy* checkbox is under *Assigned Features* > *Client Policies*.
3. Configure trust between SAP Mobile Services and SAP Business Technology Platform to enable upload of pending changes from previous users of mobile application. Refer to Configuring Security Trust for the steps to configure trust.

   > **Note**
   >
   > To test trust configuration, go to *SAP Mobile Services* > *Settings* > *Security* > *Trust* > *Test Configuration* and click *Test*. If the configuration is correct, it will display *Trust tested successfully*.



13.3.2.1 Customizing User Switch Flag

In multi-user use case, the backend needs to know when a user switch has occurred when client performs a sync.

Context

On client, tracking the current user is necessary. If the user has changed, in the subsequent sync the user switch flag must be set to true, if not, set to false.

Procedure

1. Use `OnUserSwitch` event from MDK to determine the change of user.
2. Use client API to get `OfflineDataProviderProxy`.
3. Set custom header `UserSwitch` to *True* in the `OfflineDataProviderProxy` before syncing.
4. Remove custom header `UserSwitch` when sync has completed successfully.

13.3.2.2 Customizing Multi-User Capabilities

Procedure

Logout action

1. Set *SkipReset* flag to *True* in `LogoutSkipReset.action`.

   The invocation of logout action redirects the user to the Sign-in screen. By re-entering the passcode, the user can relogin and start using the app. In a multi-user scenario, it's recommended that each user log out before they hand over the device to a different user. The default value of this flag is *False*, which means that when the app is reset, the stored credentials are deleted. It's recommended to set this flag to *True* in a multi-user scenario. The *OnUserSwitch*.js file is called when the user switches user and the Multi-User mode is enabled. All the pending Offline OData transactions from previous user are synced.
2. Call the *isAppInMultiUserMode* function to determine if the app is in single or multi-user mode.

Push Notifications

3. Unregister for push notifications during the logout action, and reregister during the *OnLoaded* event for the next user.

   When a user logs out, the user is unregistered for the push notifications. When a user signs in, the *OnLoaded* event automatically switches the push registration to the logged-in user.

13.3.2.3 Enabling Multi-User via QR Code

Context

Multiple users can log in to the same device via using a QR code.

Procedure

1. Create a QR code with CPms, and make sure that the *Allow Upload of Pending Changes from Previous User* flag is checked.

   You can navigate to *Allow Upload of Pending Changes from Previous User* from *Mobile Settings Exchange* > *Shared Devices*.
2. Set the *multiUser* flag to *True*.
3. Use the generated QR code to log in to the app.

4. Wait until the initial sync is performed. After the initial sync is finished, the *Logout* button displays in the *Profile Settings* section.
5. Tap *Logout*.

   After logging out, you can add a new user or switch between users.

Next Steps

Verify that the switch user can be done successfully with and without pending transactions.

13.4 Customizing Deep Linking

Deep linking refers to URIs that are opened outside an application, through a link from another application. When a user opens a deep link, the user is navigated to a specific resource within the application.

Deep linking configuration is performed through the metadata. See for information on working with metadata.

Deep linking supports the Create, Update, and View actions. See the following sections for objects supported and examples of deep linking schemas for the objects.

> **Note**
>
> If your mobile user is not allowed to create, update, or view an object on the client, that object is also disabled from the deep link. If your mobile user is using a persona that does not work with the object (for example, an inventory clerk persona is not configured to use work orders), the object is disabled from the deep link.

Create Action

- Schema without parameters: `samclient://create/<entitysetName>`
- Schema with parameters: `samclient://create/<entitysetName>?<parameterName1>=<parameterValue1>&<parameterName2>=<parameterValue2>...`

Following is a list of all objects supported for the create action (entitysetName, list of supported parameters), as well as an example for each object:

Create Action Objects

Object	Entity Set Name	Supported Parameters	Example
Work order	MyWorkOrderHeaders	OrderDescription PlanningPlant OrderType Priority BusinessArea MainWorkCenter MainWorkCenterPlant HeaderFunctionLocation HeaderEquipment	`samclient://create/MyWorkOrderHeaders?OrderDescription=test&MainWorkCenter=BP_ST1`
Operation	MyWorkOrderOperations	OperationShortText OrderId MainWorkCenterPlant MainWorkCenter	`samclient://create/MyWorkOrderOperations?OperationShortText=test&OrderId=4008337&MainWorkCenter=ZMECH`
Suboperation	MyWorkOrderSubOperations	OperationShortText OrderId OperationNo OperationFunctionLocation OperationEquipment	`samclient://create/MyWorkOrderSubOperations?OrderId=4008337&MainWorkCenter=BP_ST1 OperationShortText=text&OrderId=4054781&OperationNo=0040&ControlKey=CK66&MainWorkCenter=BP_ST1&OperationEquipment=10000239`
Notification	MyNotificationHeaders	NotificationDescription NotificationType HeaderFunctionLocation HeaderEquipment Priority	`samclient://create/MyNotificationHeaders?NotificationType=M1`
Timesheet	CatsTimesheets	AttendanceType ActivityType OrderId	`samclient://create/CatsTimesheets?AttendanceType=0103&ActivityType=T001&OrderId=1892368`

Object	Entity Set Name	Supported Parameters	Example
Confirmation	Confirmations	VarianceReason AccountingIndicator ActivityType OrderID PostingDate	`samclient://create/Confirmations?VarianceReason=0004&AccountingIndicator=03&ActivityType=T001&OrderID=1892368&Operation=0002&PostingDate=166730849`
Add part	MyWorkOrderComponents	OrderId OperationNo TextTypeDesc ItemCategory Plant QuantityUnE StorageLocation MaterialNum UnitOfEntry	`samclient://create/MyWorkOrderComponents`
Issue part	MaterialDocuments	Plant ItemCategory StorageLocation MaterialNum QuantityUnE WithdrawnQuantity UnitOfEntry	`samclient://create/MaterialDocuments?Plant=0001&ItemCategory=L&StorageLocation=0001&MaterialNum=ST51_R5181&QuantityUnE=4&WithdrawnQuantity=0&UnitOfEntry=eur`

Update Action

- Schema with single key and without parameters: `samclient://update/<entitysetName>/<id>`
- Schema with single key and with parameters: `samclient://update/<entitysetName>/<id>?<parameterName1>=<parameterValue1>&<parameterName2>=<parameterValue2>...`
- Schema with complex key and without parameters: `samclient://update/<entitysetName>/<keyName1>='<keyValue1>',<keyName2>='<keyValue2>'`
- Schema with complex key and with parameters: `samclient://update/<entitysetName>/<keyName1>='<keyValue1>',<keyName2>='<keyValue2>'?<parameterName1>=<parameterValue1>&<parameterName2>=<parameterValue2>..`

Following is a list of all objects supported for the create action (entitysetName, list of supported parameters). Underneath each object is a deep link example:

Update Action Objects

Object	Entity Set Name	Supported Parameters	Example
Work order	MyWorkOrderHeaders	OrderDescription Priority BusinessArea MainWorkCenter MainWorkCenterPlant HeaderFunctionLocation HeaderEquipment	`samclient://update/MyWorkOrderHeaders/4090814`
Operation	MyWorkOrderOperations	OperationShortText OperationFunctionLocation OperationEquipment	`samclient://update/MyWorkOrderOperations/OrderId='4000960',OperationNo='0180'`
Suboperation	MyWorkOrderSubOperations	OperationShortText OperationFunctionLocation OperationEquipment	`samclient://update/MyWorkOrderSubOperations/OrderId='4006511',OperationNo='0010',SubOperationNo='0011'?OperationFunctionLocation=1111-222-AA-33&OperationEquipment=10068645&OperationShortText=data`
Notification	MyNotificationHeaders	NotificationDescription HeaderFunctionLocation HeaderEquipment Priority	`samclient://update/MyNotificationHeaders/10000165?HeaderFunctionLocation=1111-222-AA-33&HeaderEquipment=10068645`
Timesheet	CatsTimesheets	AttendanceType ActivityType OrderId	`samclient://update/CatsTimesheets/lodata_sys_eid=X'F31C6D92276D427F8D6607C05179944B00000000'?ActivityType=TRAVEL&OrderId=4008338`
Confirmation	Confirmations	VarianceReason AccountingIndicator ActivityType	`samclient://update/`

Object	Entity Set Name	Supported Parameters	Example
Add part	MyWorkOrderComponents	OrderId QuantityUnE StorageLocation MaterialNum UnitOfEntry	`samclient://update/`

View Action

- Schema to view list screen: `samclient://view/<entitysetName>`
- Schema to view details screen: `samclient://view/<entitysetName>/<id>` or `samclient://view/<entitysetName>/<keyName1>='<keyValue1>',<keyName2>='<keyValue2>'`

Following is a list of all objects supported for the create action (entitysetName, list of supported parameters). Underneath each object is a deep link example:

View Action Objects

Object	Entity Set Name	Supported Parameters	Example
Work order	MyWorkOrderHeaders		`samclient://view/MyWorkOrderHeaders/4007187`
Operation	MyWorkOrderOperations		`samclient://view/MyWorkOrderOperations/OrderId='4008338',OperationNo='0010'`
Suboperation	MyWorkOrdeSubOperations		`samclient://view/MyWorkOrderSubOperations/OrderId='4008337',OperationNo='0010',SubOperationNo='L001'`
Notification	MyNotificationHeaders		`samclient://view/MyNotificationHeaders/10020862`
Equipment	MyEquipments		`samclient://view/MyFunctionalLocations/FLOC-SVB-AN-07`
Functional location	MyFunctionalLocations		`samclient://view/MyEquipments/10038476`

14 Data Distribution Model Overview

A data distribution model defines how and what back end data are downloaded to the mobile devices.

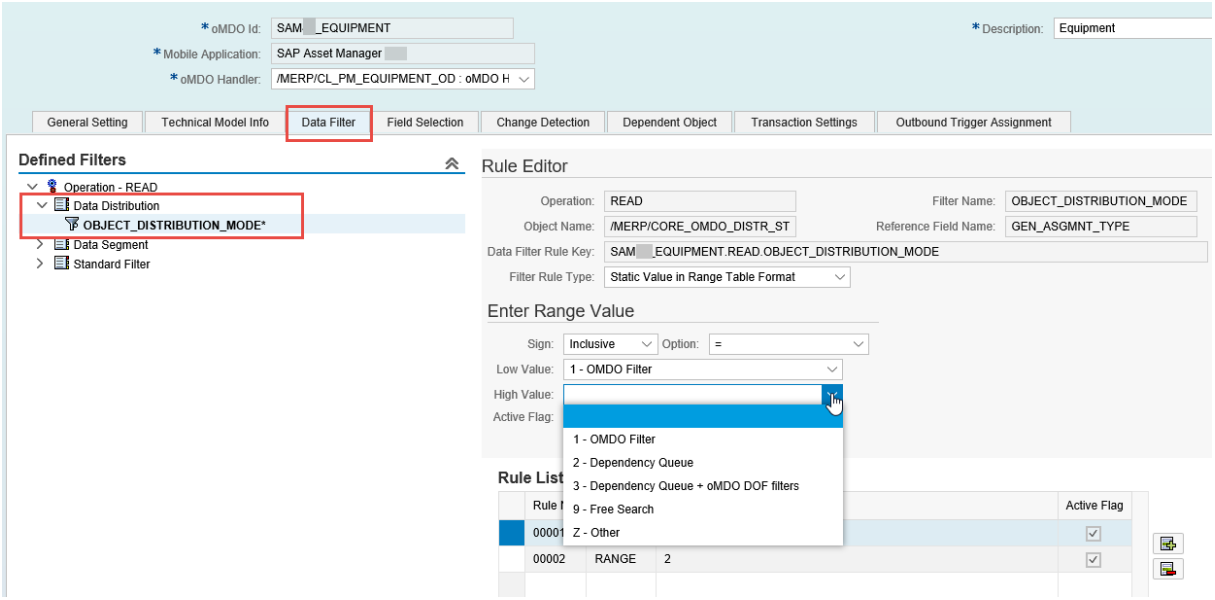
Data distribution models consider various factors when determining what backend data should be downloaded to the mobile client and to the mobile user. Some common criteria are:

- Relevant business processes and business rules
- User business roles in the organization
- Last mobile client synchronization time with the back-end system
- Backend data changes and the time of the changes
- Current state of data objects on the mobile device

For the initial synchronization from the mobile device to the back-end system, the first two bullet points are considered when determining what data should be downloaded to the mobile device and for the requesting user. For subsequent delta synchronizations from the mobile device to the back-end system, all bullet points are considered when determining what data should be downloaded to the mobile device for the requesting user.

The following data distribution models are supported for the SAP Service and Asset Manager application:

- **OMDO Filters**
  Object data collection entirely depends on *OMDO* filter conditions.
- **Dependency Queue**
  Object data collection entirely depends on Dependency Queue objects, and no filter conditions are applied for the fetch criteria.
- **Dependency Queue + OMDO DOF Filters**
  Object data collection is based on dependency queue objects, and the *OMDO DOF* filters are applied for the result set.
- **Other (Custom BAdI)**
  You can implement your own distribution logic using a BAdI.



By default, the SAP Service and Asset Manager application determines object data collection based on distribution model by using *OMDO* filters. In some cases, both *OMDO* filters and *Dependency Queue* are used depending on the business use case. You can choose the appropriate distribution model based on your specific business processes and requirements.

To change the data distribution model for a particular *OMDO* object, complete the steps below:

1. On the ConfigPanel home page, choose *OData Mobile Data Object Configuration*.
   Make sure you select your desired mobile application in the *Mobile Application Filter* field at the top of the page.
2. From the *OData Mobile Data Object List* select the desired OMDO object, such as *SAM2310_EQUIPMENT*, and then click on the *Data Filter* tab.
3. Expand the *Defined Filters* list under *Operation - READ* > *Data Distribution* > *OBJECT_DISTRIBUTION_MODE*. Choose the *Change* button from the menu.
4. Set the distribution model.
5. Save your changes.

14.1 Data Distribution by Assignment Types

By default, the SAP Service and Asset Manager application determines the assignment of work orders and notifications using the personnel number assignment at header level. However, implementation environments in different industries or business types may use a different assignment model from the default to determine the proper technician assignment for work orders and notifications. The SAP Service and Asset Manager application supports several assignment models; you only need to change the assignment type configuration for the specific model.

See Business Object Distribution by Assignment Model [page 152] for more details about assignment model distribution, and how to change assignment type for both work order and notification.

14.2 Common Filters for SAP Service and Asset Manager

The filters listed in the following table are common to all SAP Service and Asset Manager distribution rules. See the specific rules for details on filter requirements for those rules.

Common Filters: Back End Data Filter Matrix

Common Filters: Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution, Mandatory	See specific rule for value	Defines which distribution model is used
COMP_CODE	Data Distribution, Optional	Any	Restricts work order distribution based on the maintenance plant company code on the work order.
CO_AREA	Data Distribution, Optional	Any	Restricts work order distribution based on work order maintenance plant controlling area.
DATE_CLOSE	Standard Filter, Optional	Any	Restricts work order distribution based on date range within which work order has been closed.
DATE_COMPLETION	Standard Filter, Optional	Any	Restricts work order distribution based on date range within which work order has been technically completed.
DATE_RELEASE	Standard Filter, Optional	Any	Restricts work order distribution based on date range within which work order has been released
ORDER_CATG	Data Distribution, Optional	See specific rule for value	Restricts work order distribution based on work order category. For maintenance orders, it should be value `30`.
ORDER_TYPE	Data Distribution, Optional	Any	Restricts work order distribution based on work order type.
ORDERID	Data Distribution, Optional	Any	Restricts work order distribution to a specific work order number.
OPER_ACTTYPE	Standard Filter, Optional	Any	Restricts work order distribution based on work order operation level activity type.
OPER_CONTROL_KEY	Standard Filter, Optional	Any	Restricts work order distribution based on the operation level control key on the work order.

Filter Name	Type	Value	Comments
OPER_EXCL_SYST_STAT	Standard Filter, Optional	Any	Restricts work order distribution by excluding work orders with operations that have a specified system status code.
OPER_EXCL_USER_STAT	Standard Filter, Optional	Any	Restricts work order distribution by excluding work orders with operations that have a specified user status code.
OPER_INCL_SYST_STAT	Standard Filter, Optional	Any	Restricts work order distribution by requiring work order operations with a specified system status code.
OPER_INCL_USER_STAT	Standard Filter, Optional	Any	Restricts work order distribution by requiring work order operations with a specified user status code.
OPER_PLANT	Standard Filter, Optional	Any	Restricts work order distribution based on the operation level plant of the work order.
PLANNER_GROUP	Data Distribution, Optional	Any	Restricts work order distribution based on the header planner group of the work order.
PLANNING_PLANT	Data Distribution, Optional	Any	Restricts work order distribution based on the work order header planning plant.
PLANT	Data Distribution, Optional	Any	Restricts work order distribution based on the header maintenance plant of the work order.
PM_PHASE	Data Distribution, Optional	Any	Restricts work order distribution based on the maintenance process phase of the work order.
WO_EXCL_SYST_STAT	Standard Filter, Optional	Any	Restricts work order distribution by excluding work orders header with a specified system status code.
WO_INCL_SYST_STAT	Standard Filter, Optional	Any	Restricts work order distribution by requiring a work order header with a specified system status code.
WO_EXCL_USER_STAT	Standard Filter, Optional	Any	Restricts work order distribution by excluding a work order header with a specified user status code.
WO_INCL_USER_STAT	Standard Filter, Optional	Any	Restricts work order distribution by requiring a work order header with a specified user status code.
MAIN_WORK_CENTER	Data Distribution, Optional	Any	Restricts work order distribution based on the work order header work center.

Filter Name	Type	Value	Comments
DOC_GOS_RELTYPE	Standard Filter, Optional	Data Segment, Optional	Determines whether the GOS attachment is supported based on a GOS relationship.
DMS_DOC_TYPE	Standard Filter, Optional	Data Segment, Optional	Determines whether the DMS attachment is supported based on the DMS document type.
DOC_LINK_OBJ	Standard Filter, Optional	Data Segment, Optional	Determines whether the DMS attachment is supported based on the linked SAP object.

14.3 User Parameter IDs

User Parameter IDs are used to filter data that mobile users download when they perform a sync. By setting the Parameter IDs for users, you can define which assignment types technicians will receive and which activities they have to complete on their mobile devices. To update these IDs in your user profile, perform the following steps:

1. Use the transaction code *SU3* on the SAP GUI.
2. Select the *Parameters* tab.
3. In the list of parameters you can add a new row or modify/delete existing ones.

> **Note**
>
> - To display other users' Parameter IDs, use the transaction code *SU01D* on the SAP GUI.
> - You need authorization to edit the list of User Parameter IDs for other users.

List of User Parameter IDs

The following table lists all User Parameter ID values available in SAP Service and Asset Manager:

User Parameter ID	Description
AGR	Work center
SWK	Maintenance plant
VAP	Main work center for maintenance tasks
WRK	Plant
IWK	Maintenance planning plant
CAC	Controlling area
IHG	Maintenance planner group

User Parameter ID	Description
KOS	Cost Center
LAG	Storage location
PBR	Personnel Area
RVK	Reference sales organization

> **Note**
>
> For more information about which assignment types require User Parameter IDs, see:
>
> - Work Order Assignment Type Options [page 153]
> - Notification Assignment Options [page 154]

14.4 Work Order Distribution Rules

The standard SAP Service and Asset Manager application work order distribution is controlled by the OMDO (OData mobile data object) **SAM2310_WORK_ORDER_GENERIC** *READ* operation. It supports several data distribution models for the work order.

You can choose the appropriate distribution model based on your specific business processes and requirements.

14.4.1 Distribution by Work Order Header Person Responsible

Requirements

The following are requirements before configuring the distribution model for Distribution by Work Order Header Person Responsible:

- Mobile user has an employee number (personnel number) assigned in SAP
- Employee number is assigned to the work order header as the person responsible
- Work order has been released
- Work order has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	1	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, it should be value `30`.

14.4.2 Distribution by Work Order Operation Person Responsible

Requirements

The following are requirements before configuring the distribution model for Distribution by Work Order Header Person Responsible:

- Mobile user (i.e., the technician) must have an employee number (personnel number) assigned in SAP
- Employee number is assigned to the work order operation as the person responsible
- Work order is released
- Work order is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	2	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.4.3 Distribution by Work Order Suboperation Person Responsible

Requirements

The following are requirements before configuring the distribution model for Distribution by Work Order Suboperation Person Responsible:

- Mobile user (i.e., the technician) must have an employee number (personnel number) assigned in SAP
- Employee number is assigned to the work order suboperation as the person responsible
- Work order is released
- Work order is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	3	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.4.4 Distribution by Capacity Requirement Person Responsible

Requirements

The following are requirements before configuring the distribution model for Distribution by Capacity Requirement Person Responsible:

- Mobile user (i.e., the technician) must have an employee number (personnel number) assigned in SAP
- Employee number is assigned to the work order capacity requirement split records as the person responsible
- Work order is released
- Work order is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	4	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.4.5 Distribution by Work Order Header Planner Group

Requirements

The following are requirements before configuring the distribution model for Distribution by Work Order Header Planner Group:

- Mobile user (i.e., the technician) has been assigned to the planner group based on the business
- Employee number is not required
- Planner group associated with the mobile user is assigned to the work order header
- Work order is released
- Work order is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	5	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.4.6 Distribution by Work Order Operation Work Center

Requirements

The following are requirements before configuring the distribution model for Distribution by Work Order Operation Work Center:

- Mobile user (i.e., the technician) has been associated with a work center in business
- Employee number is not required
- Work center associated with the mobile user is assigned to work order operation
- Work order has been released
- Work order has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	6	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.4.7 Distribution by Work Order Header Business Partner

Requirements

The following are requirements before configuring the distribution model for Distribution by Work Order Header Business Partner:

- Predefined partner function is used in work order for work assignment
- Employee number is not required
- By default, the SAP user ID for the mobile user is used for data distribution by partner function
- Work order is released
- Work order is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	7	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.4.8 Distribution by Work Order Header Work Center

Requirements

The following are requirements before configuring the distribution model for Distribution by Work Order Header Work Center:

- Mobile user (i.e., technician) has been associated with a work center based on the business
- Employee number is not required
- Work center associated with the mobile user is assigned to the work order header
- Work order is released
- Work order is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	8	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.4.9 Distribution through MRS Scheduling Engine

Requirements

The following are requirements before configuring the distribution model for Distribution through MRS Scheduling Engine:

- MRS has been implemented in the SAP system, and is responsible to schedule and update work order capacity records with the assigned technician
- Employee number is required for the mobile user
- Work order is released
- Work order is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	A	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.4.10 Distribution by Free Search

Requirements

The following are requirements before configuring the distribution model for Distribution by Free Search:

- Free search criteria for the work order. Used for an OnDemand work order look-up scenario.
- Employee number is not required
- Work order is released
- Work order is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
WO_ASSIGNMENT_TYPE	Data Distribution Mandatory	7	Defines which distribution model is used
ORDER_CATG	Data Distribution Optional	30	Restricts work order distribution based on work order category. For maintenance orders, value should be *30*.

14.5 Notification Distribution Rules

The OMDO (OData mobile data object) **SAM2310_NOTIF_ASSIGNMENT_TYPE** *READ* operation controls the standard SAP Service and Asset Manager application notification distribution. It supports several data distribution models for the notification.

You can choose the appropriate distribution model based on your specific business processes and requirements.

Notification requests are assigned to the technician directly or assigned through the work center, planner group, or related business partner of the technician. The SAP Service and Asset Manager application supports these different assignment types while downloading notifications associated with the technician.

Valid assignment types are:

- **1 - Header Level Person Responsible:** Assign this notification to the HR personnel number of the technician through the notification header *Partner* section.
- **2 - Notification Task Level Personnel Number:** Assign this notification to the HR personnel number of the technician through individual *Task Personnel Number* field.
- **3 - Header Level Planner Group:** Assign this notification to the planner group associated with the technician through the header level *Planner Group* field.
- **4 - Header Level Business Partner:** Assign this notification to the business partner associated with the technician through header level *Partner Function Maintenance*. The business partner can be anyone related to the notification partner function and associated with the technician, such as user responsible, sold-to-party, or other party. If there is no MAM configuration set up for the user, the default configuration uses *VU-User Responsible* as the default partner function and the technician *SAP User ID* as the partner number.
- **5 - Header Level Work Center:** Assign this notification to the work center associated with the technician through the header level *Work Center* field.
- **D - Dependency Queue:** Enables the dependency queue from the work order. When active, all notifications associated with a work order are downloaded, as well as qualifying data based on additional distribution rules that are set.

Customers can choose the appropriate distribution model based on their specific business processes and requirements.

14.5.1 Distribution by Notification Header Person Responsible

Requirements

The following are requirements before configuring the distribution model for Distribution by Notification header Person Responsible:

- Technician has an employee number (personnel number) assigned in SAP
- Employee number is assigned to the notification header partner overview as the person responsible
- Notification is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
NOTIF_ASSIGNMENT_TYPE	Data Distribution Mandatory	1	Defines which distribution model is used

14.5.2 Distribution by Notification Task Level Person Responsible

Requirements

The following are requirements before configuring the distribution model for Distribution by Notification task level Personal Responsible:

- Technician has an employee number (personnel number) assigned in SAP
- Employee number is assigned to the notification task *Person Responsible* field
- Notification is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
NOTIF_ASSIGNMENT_TYPE	Data Distribution Mandatory	2	Defines which distribution model is used

14.5.3 Distribution by Notification Header Level Planner Group

Requirements

The following are requirements before configuring the distribution model for Distribution by Notification header level Planner Group:

- Technician is assigned to the planner group based on the business
- Employee number is not required
- Planner group associated with the mobile user is assigned to the notification header
- Notification is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
NOTIF_ASSIGNMENT_TYPE	Data Distribution Mandatory	3	Defines which distribution model is used

14.5.4 Distribution by Notification Header Level Business Partner

Requirements

The following are requirements before configuring the distribution model for Distribution by Notification header level Business Partner:

- Business partner is assigned to the notification header partner overview
- The SAP user ID of the mobile user is used for data distribution by partner function using *User Responsible*

- Employee number is not required
- Notification is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
NOTIF_ASSIGNMENT_TYPE	Data Distribution Mandatory	4	Defines which distribution model is used

14.5.5 Distribution by Notification Header Level Work Center

Requirements

The following are requirements before configuring the distribution model for Distribution by Notification header level Work Center:

- Technician is associated with a work center in business
- Employee number is not required
- Work center associated with the mobile user is assigned to the notification header
- Notification is not marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
NOTIF_ASSIGNMENT_TYPE	Data Distribution Mandatory	5	Defines which distribution model is used

14.5.6 Distribution by Free Search - Notifications

Requirements

The following are requirements before configuring the distribution model for Distribution by Free Search:

- Free search for notification used for an on-demand notification look-up scenario

- Employee number is not required
- Notification is not marked for deletion

Back End Data Filter Matrix

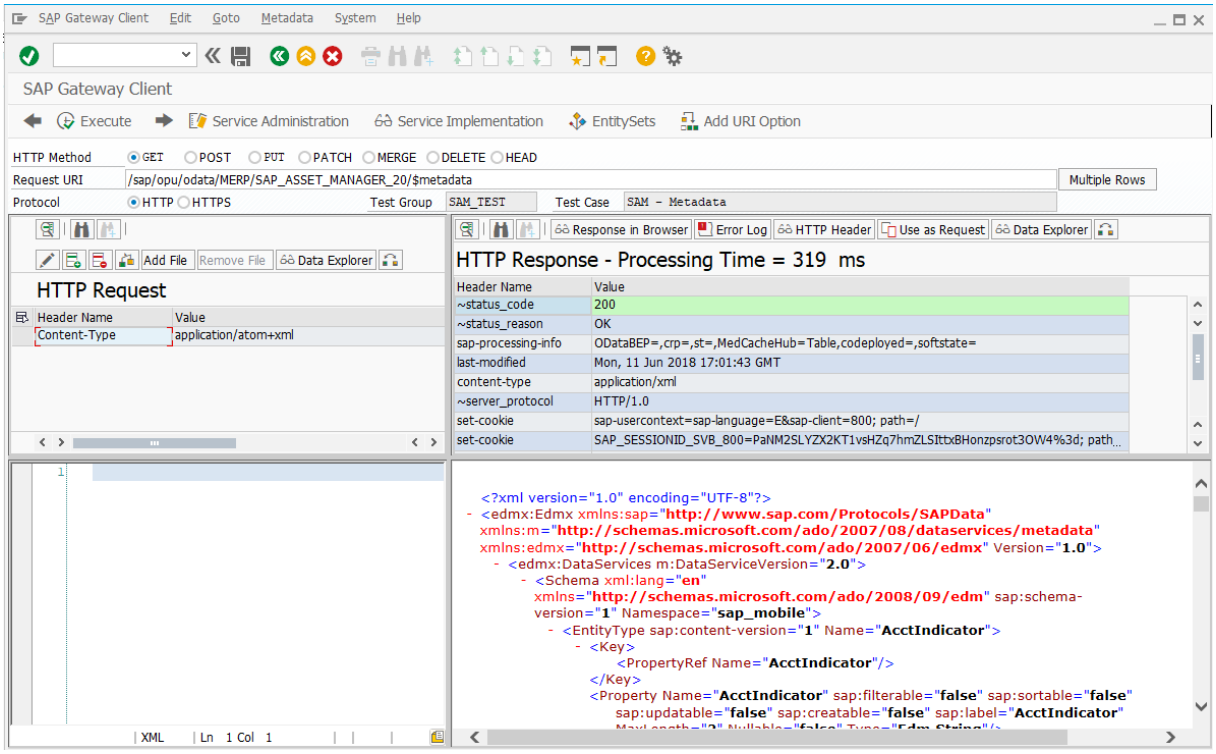
Filter Name	Type	Value	Comments
NOTIF_ASSIGNMENT_TYPE	Data Distribution Mandatory	9	Defines which distribution model is used

15 Troubleshooting

This section describes the various troubleshooting activities that you can perform in error situations, or the app users can perform on a regular basis to ensure the smooth running of the mobile application. It is also explains how to monitor the different components of SAP Gateway, how to use the logs, and how to carry out maintenance activities.

15.1 SAP Gateway Client

You can use the SAP Gateway Client (transaction code: */IWFND/GW_CLIENT*) to test your OData service provider without an OData consumer, such as the SAP Service and Asset Manager mobile client. This tool is especially useful to test your OData service from the back end to identify service-related issues before a service is used by the mobile application.



For more information about how to work with the SAP Gateway Client, see SAP Gateway Client in the SAP Gateway Technical Operations Guide.

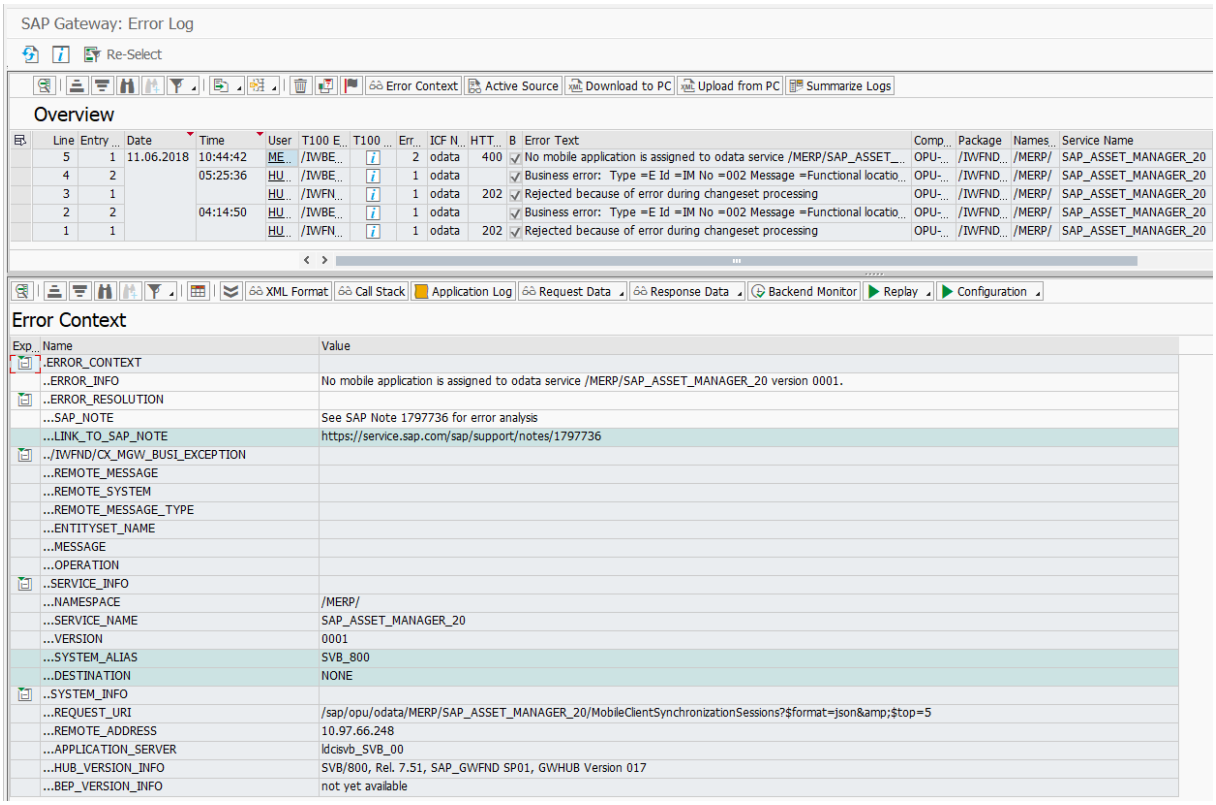
15.2 SAP Gateway Error Logs

Error logs provide detailed context information about errors that have occurred at runtime, enabling you to perform root cause analysis, as well as reproducing and correcting errors.

You can launch the error log with transaction */IWFND/ERROR_LOG* in Gateway Hub systems. Launch the error log with transaction */IWBEP/ERROR_LOG* in your back-end system.

The SAP Gateway error logs reveal basic details about errors and show errors from all users for a given client. Business logic errors are often displayed in this error log due to improper business logic. Other errors displayed include the HTTP code to indicate the type of error.

Note that based on the security level setting, advanced details or the replay function may be hidden or disabled. Note also that these error logs will not show generic authorization errors if users fail to properly authenticate.



You can navigate to different sections from the *Error Context* area as shown above. Choose *Replay* to reproduce and correct errors. Choose from the following two replay options:

- SAP Gateway Client
- Web Browser

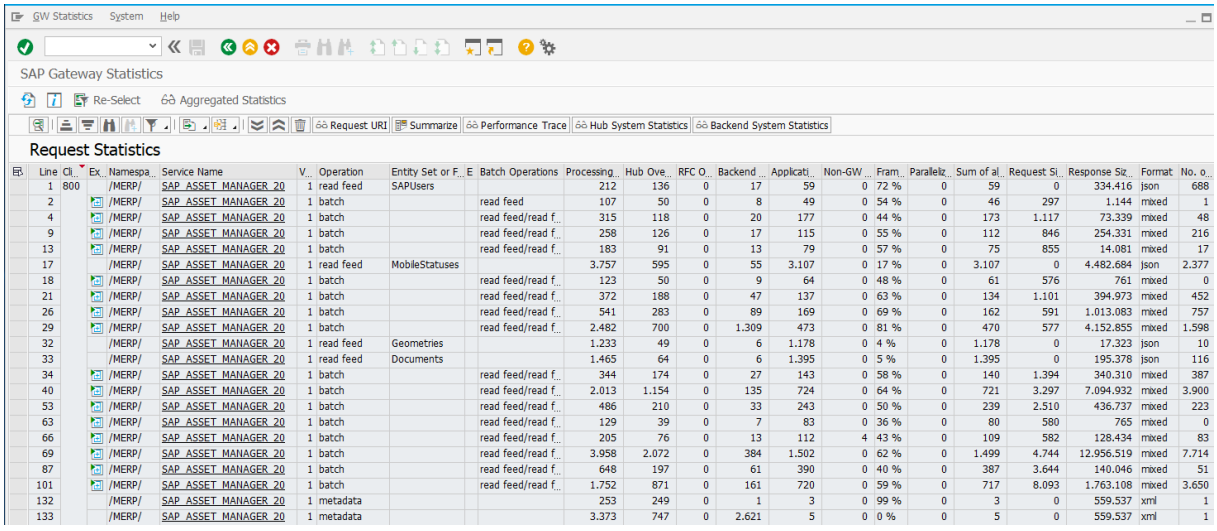
Use option *SAP Gateway Client* to reproduce runtime situations that led to a particular error without accessing the application from the actual mobile client, and to simulate a service at runtime to identify and resolve potential issues.

For more information about how to configure the error log, see Configuration Settings for the Error Log in the *SAP Gateway Technical Operations Guide*.

In addition, use the *Application Log Viewer* to display more technical error details by using transaction */IWFND/APPS_LOG*.

15.3 SAP Gateway Statistics

You can use the SAP Gateway Statistics (transaction code: */IWFND/STATS*) to display the request statistics and aggregated statistics. Each successful OData request has an entry in the statistics records, which is kept for 7 days by default, however, you can extend the period to 30 days. Request statistics can be aggregated, in which case they are kept for 90 days by default, however, you can extend the period to 365 days.



Line	Cli..	Ex..	Namespa..	Service Name	V..	Operation	Entity Set or F..	E	Batch Operations	Processing..	Hub Ove..	RFC O..	Backend ..	Applicati..	Non-GW ..	Fram..	Paralleliz..	Sum of al..	Request Si..	Response Siz..	Format	No. o..
1	800		/MERP/	SAP_ASSET_MANAGER_20	1	read feed	SAPUsers			212	136	0	17	59	0	72 %	0	59	0	334.416	json	688
2			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed	107	50	0	8	49	0	54 %	0	46	297	1.144	mixed	1
4			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	315	118	0	20	177	0	44 %	0	173	1.117	73.339	mixed	48
9			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	258	126	0	17	115	0	55 %	0	112	846	254.331	mixed	216
13			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	183	91	0	13	79	0	57 %	0	75	855	14.081	mixed	17
17			/MERP/	SAP_ASSET_MANAGER_20	1	read feed	MobileStatuses			3.757	595	0	55	3.107	0	17 %	0	3.107	0	4.482.684	json	2.377
18			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	123	50	0	9	64	0	48 %	0	61	576	761	mixed	0
21			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	372	188	0	47	137	0	63 %	0	134	1.101	394.973	mixed	452
26			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	541	283	0	89	169	0	69 %	0	162	591	1.013.083	mixed	757
29			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	2.482	700	0	1.309	473	0	81 %	0	470	577	4.152.855	mixed	1.598
32			/MERP/	SAP_ASSET_MANAGER_20	1	read feed	Geometries			1.233	49	0	6	1.178	0	4 %	0	1.178	0	17.323	json	10
33			/MERP/	SAP_ASSET_MANAGER_20	1	read feed	Documents			1.465	64	0	6	1.395	0	5 %	0	1.395	0	195.378	json	116
34			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	344	174	0	27	143	0	58 %	0	140	1.394	340.310	mixed	387
40			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	2.013	1.154	0	135	724	0	64 %	0	721	3.297	7.094.932	mixed	3.900
53			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	486	210	0	33	243	0	50 %	0	239	2.510	436.737	mixed	223
63			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	129	39	0	7	83	0	36 %	0	80	580	765	mixed	0
66			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	205	76	0	13	112	4	43 %	0	109	582	128.434	mixed	83
69			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	3.958	2.072	0	384	1.502	0	62 %	0	1.499	4.744	12.956.519	mixed	7.714
87			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	648	197	0	61	390	0	40 %	0	387	3.644	140.046	mixed	51
101			/MERP/	SAP_ASSET_MANAGER_20	1	batch			read feed/read f..	1.752	871	0	161	720	0	59 %	0	717	8.093	1.763.108	mixed	3.650
132			/MERP/	SAP_ASSET_MANAGER_20	1	metadata				253	249	0	1	3	0	99 %	0	3	0	559.537	xml	1
133			/MERP/	SAP_ASSET_MANAGER_20	1	metadata				3.373	747	0	2.621	5	0	0 %	0	5	0	559.537	xml	1

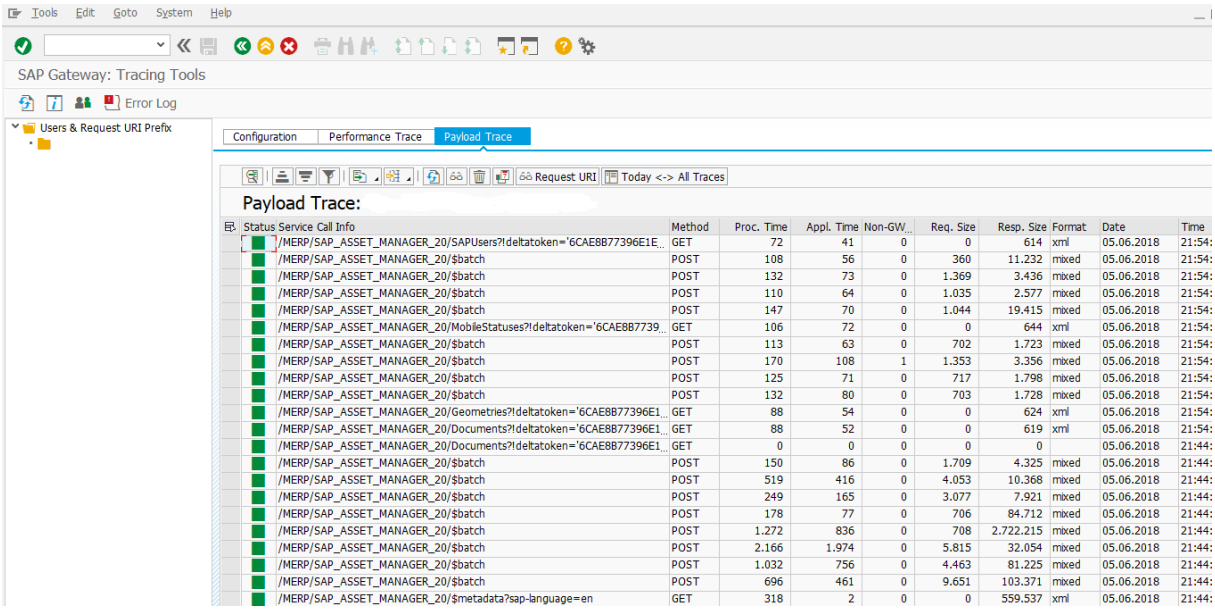
SAP Gateway Statistics aggregates the entries by various entities, for example, client, namespace, service name and version. With the */IWFND/STATS* transaction you can verify details, such as processing time, response size by entity, and other statistics about the complete request.

15.4 SAP Gateway Tracing Tools

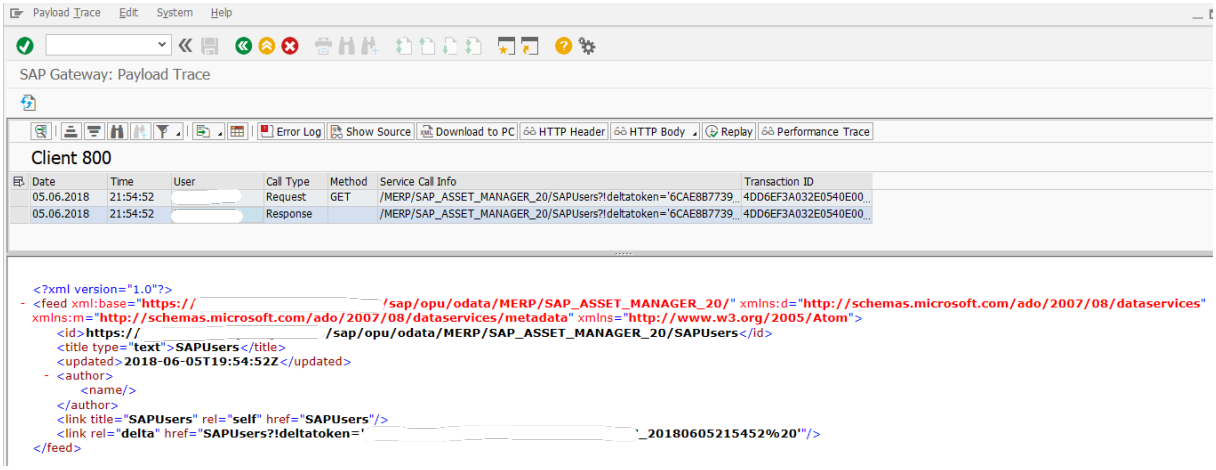
The SAP Gateway provides tracing tools (transaction code: `/IWFND/TRACES`) to trace on a particular user for both performance and payload.

Performance trace enables you to monitor performance at service call level for both the SAP Business Suite and the SAP Gateway. Payload trace enables you to monitor the service calls with request and response data, and to replay and simulate the service calls without accessing the application from the mobile client.

Traces display detailed request and response data coming into the SAP Gateway. Traces are active for only a short time, and are purged on a regular basis.



With this tool, you can verify the exact content of the request header and body that is sent from the mobile device, and also check the response from the SAP Gateway.



For information about how to configure and activate the payload trace tool, see Tracing Tools: Configuration in the *SAP Gateway Technical Operations Guide*.

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

© 2024 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.

THE BEST RUN SAP