



PUBLIC

SAP Work Manager

Document Version: 6.5.0 – 2019-04-18

SAP Work Manager Configuration Guide

Content

- 1 SAP Work Manager Overview. 6**
- 1.1 SAP Work Manager Functional Overview. 6
 - SAP ERP Transaction Codes for SAP Work Manager. 9
- 1.2 SAP Work Manager on the SAP Mobile Platform. 10
- 1.3 Setting up the SAP Mobile Platform Server to Support Multiple Languages. 12
- 1.4 Multiple Country Support for Address and Phone Formats. 13
- 2 Mobile Add-On for ERP Configuration Panel. 19**
- 2.1 Accessing the Mobile Add-On for ERP ConfigPanel. 19
- 2.2 ConfigPanel Overview. 20
 - Standard Operations in the Configuration Panel. 20
 - ConfigPanel - General Info Section. 21
- 2.3 Mobile Add-On Configuration Panel. 33
 - Configuration Panel Overview. 34
 - ConfigPanel - Technical Settings. 38
 - ConfigPanel - Geospatial Service Definitions Settings. 39
 - Data Store Definition. 39
 - Data Agent Definition. 40
 - Inbound Transaction Queue Definition. 40
 - ConfigPanel - Mobile Application Configuration Settings. 42
 - Push Scenario Definition. 50
 - Subscription Agent Definition. 51
 - ConfigPanel - Exchange Object Configuration Settings. 51
 - ConfigPanel - EFI Assignment Settings. 56
 - Mobile Data Object Configuration. 58
 - BAPI Wrapper Configuration. 62
 - Outbound Trigger Configuration. 64
 - ConfigPanel - Security Settings. 64
- 3 Data Distribution Model Overview. 71**
- 3.1 Common Filters. 71
- 3.2 Work Order Distribution Rules. 74
 - Distribution by Work Order Header Person Responsible. 75
 - Distribution by Work Order Operation Person Responsible. 76
 - Distribution by Work Order Sub-Operation Person Responsible. 77
 - Distribution by Capacity Requirement Person Responsible. 78
 - Distribution by Work Order Header Planner Group. 79

	Distribution by Work Order Operation Work Center.	80
	Distribution by Work Order Header Business Partner.	81
	Distribution by Work Order Header Work Center.	82
	Distribution through MRS Scheduling Engine.	83
	Distribution by Free Search.	83
	Distribution by Custom Logic.	84
3.3	Notification Distribution Rules.	85
	Distribution by Notification Header Person Responsible.	85
	Distribution by Notification Task Level Person Responsible.	86
	Distribution by Notification Header Level Planner Group.	87
	Distribution by Notification Header Level Business Partner.	88
	Distribution by Notification Header Level Work Center.	88
	Distribution by Free Search (Notifications).	89
	Distribution by Custom Logic (Notifications).	90
3.4	Periodic Meter Reading Distribution Rules.	91
	Distribution by Free Search (Periodic Meter Readings).	91
	Distribution by Meter Reading Unit.	92
	Distribution by Personnel Number.	93
	Distribution by Business Partner.	93
	Distribution by Enhancement Spot Implementation.	94
4	Configuring SAP Work Manager with LAM.	96
4.1	Disabling and Re-Enabling Linear Asset Management.	96
4.2	Adding a LAM Equipment Type Filter.	97
4.3	Adding Linear Data for Characteristics.	98
5	Configuring SAP Work Manager with Customer Service.	100
6	Configuring SAP Work Manager with Meter Management.	101
7	Common Changes for Work Manager.	102
7.1	Copying an Object to the Customer Namespace.	102
7.2	Working with BAPI Wrappers.	103
	Changing the MDO Assignment of a BAPI Wrapper.	104
7.3	Working with MDO Filter Rules.	104
	Changing MDO Filter Rules.	105
7.4	Work Order Distribution by Order Type.	106
	Configuring Work Order Distribution by Order Type.	106
7.5	Filtering Equipment Complex Table Records by Equipment Status.	108
	Configuring an Equipment Status Filter for an Equipment Complex Table.	109
7.6	Work Order Assignment Type Options.	110
7.7	Work Order STARTED Status Mapping in SAP ERP Overview.	112
	Change the Mapping of Mobile Status STARTED.	112

7.8	Retrieving Additional Data for Mobile Data Objects Overview.	114
	Adding New Values to Retrieve for Mobile Data Objects.	114
7.9	Enabling the Technical Objects Work List Feature.	117
7.10	Working with Rounds.	119
7.11	Configuring Multi-Resource Scheduling (MRS).	120
7.12	Disabling and Re-Enabling Attachments.	121
7.13	Working with Pushes.	124
	Configuring Push.	124
	Configuring Push with the SAP Cloud Platform, Mobile Service.	130
	Push Scenarios for Operations and Suboperations Overview.	131
7.14	Integrating GIS with the SAP Work Manager Application.	133
7.15	Working with Inbound Transactions.	134
	Enabling Inbound Transaction Management.	137
	Inbound Transactions - Client Object Locking Matrix.	138
	Inbound Queues in SAP Work Manager.	141
	Key Mapping Processing.	144
	Inbound Transaction Processing: Review and Approval Workflow.	146
	Developing a Custom Inbound Queue Handler.	147
7.16	Data Staging.	147
7.17	Assigning Responsible Vehicles to Users.	149
8	Upgrade Considerations.	152

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/viewer/product/SAP_Work_Manager/.

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

Version	Date	Description
6.5.0	DEC 2017	Original release
6.5.1	SEP 2018	Added DMS limitation for equipment and functional location in Disabling and Re-Enabling Attachments [page 121]
6.5.2	APR 2019	Revised the note in the topic Push Scenarios for Operations and Suboperations Overview [page 131]

1 SAP Work Manager Overview

The *SAP Work Manager Configuration Guide* is intended for system administrators, technical architects, 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 mobile setup of SAP.

Use the *SAP Work Manager Configuration Guide* along with appropriate SAP documentation. Note that this guide only covers setting up and enabling the SAP Work Manager mobile application.

1.1 SAP Work Manager Functional Overview

With SAP Work Manager, employees have SAP ERP data available, including task lists, repair histories, and material availability.

SAP Work Manager is a comprehensive mobile solution for the automation of asset management and field service processes. Regardless of connectivity, SAP Work Manager allows remote employees to access, complete, and manage their assigned work orders and service requests through mobile devices.

SAP Work Manager comes packaged with the Mobile Add-On for ERP. Because an SAP ABAP add-on installs into the SAP namespace in your SAP system, the Mobile Add-On for ERP offers tight integration and easy deployment. It also does not interfere to or from your existing SAP system customizations or standard SAP objects. The Mobile Add-On for ERP allows for full configuration, management, and visibility into your SAP Work Manager application from within your SAP system infrastructure.

The main features and functions available in SAP Work Manager include the following:

Work Orders and Notifications

SAP Work Manager manages work orders by extending SAP ERP functionality through the following actions:

- Displaying work orders and their details
- Editing work orders and their details
- Creating local work orders and their details

SAP Work Manager uses the SAP ERP notification functionality to push notifications to a mobile device. From there, users can perform the following actions:

- Display and edit notification details
- Create and edit existing notifications
- Delete existing notifications

Work orders and notifications are used for:

- Push and monitoring
- Viewing both historical work orders and pending work orders
- Location and equipment
- Short and long texts
- Operations and suboperations
- Locally created (user created) work orders and notifications
- Tasks, activities, and items
- Status mapping and filters

Data capture

The client can capture data from the following items, and passes the data to the back end during a transmit:

- Time and attendance
- Materials used
- Damage codes
- PM confirmations

Time

SAP Work Manager extends the necessary time tracking functionality in SAP ERP to the mobile device so field personnel can track their time efficiently and accurately.

Single Sign-On (SSO)

Single Sign-On (SSO) allows the user to log into the SAP Work 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, the mobile user can access another mobile application without the need to log in again.

SSO gives the mobile user the ability to change or reset the logon password from the mobile device.

Linear Asset Management (LAM)

Linear Asset Management provides functions with which you can describe, display, and manage linear assets.

Linear assets are technical systems with a linear infrastructure whose condition and properties can vary from section to section (dynamic segmentation).

Examples of linear assets include:

- Pipelines
- Roads and railway tracks
- Overhead power lines and cables

You can create linear assets as technical objects (such as functional locations and equipment) and store linear data. You can carry out maintenance tasks for these technical objects, which result in notifications, work orders, and measurement documents. Therefore, you can monitor what condition your linear assets are in, identify where there is damage or a defect, and manage all types of maintenance tasks.

Inspection Rounds

Inspection rounds have the following characteristics:

- An inspection round is planned, for example, once a week, once a day, or for each shift
- An inspection round is always completed within one day, or within one shift
- The duration of an inspection round includes the work and the round itself within one plant
- Work is done on different technical objects. The number of technical objects can vary, depending on the sequence and the time required.
- The work activities in an inspection round are similar and require the same tools, replacement parts, and qualifications
- You can assign an inspection round to either a work center or to a person
- Measuring points on an inspection round are displayed in an easy to use wizard format
- If required, the user can create measurement documents from the inspection round
- If follow up activities are necessary, the user can create notifications from the inspection round

An inspection round can involve the following objects:

- Functional locations
- Equipment
- Measuring points

Confirming the inspection round is a key element of the overall inspection round process and is easy for the mobile user. The technical confirmation is done in a new entry transaction. All orders of type **INSP** and all orders that have **PRT** assigned to the equipment or functional locations at an Operation level are classified as *Inspection Round* work orders on the mobile client.

Rounds

Rounds is integrated with the SAP Work Manager application, and groups measuring points together.

When you download either an entry list or a single measurement point, you are given all the information you need for that entry list or point. The information you are given includes the equipment number, unit of measure, upper and lower limits for the measuring point, and any measurement history. You can perform your rounds by moving down your entry list, taking each reading, and recording any observations as you go.

Rounds lets you enter measuring points. In SAP ERP, a measuring point is assigned to either an entry list, a functional location, or a piece of equipment. Each entry list, functional location, or piece of equipment can have

one, two, or multiple measuring points. Rounds retrieves the collection of measuring points assigned to an entry list, functional location, or piece of equipment and presents them so you can enter readings for one or more of the measuring points in the collection. You can also retrieve individual measuring points to take their readings. Each collection is assigned a unique name and is displayed under the Rounds Detail screen in the SAP Work Manager application.

Attachments

SAP Work Manager supports viewing of master data or transaction data attachments on the mobile device. Attachments include Microsoft Office files, PDF files, and other commonly used business files, including videos, pictures, and audio. When you select [Attachments](#), the Details screen displays the attachments that are available for download.

By default, the option to download attachments is on, but you can configure the option through the Configuration Panel based on work order type. If the work order has any attachments, only limited information regarding the attachments is fetched during an initial sync. The attached content is downloaded to the device via push process only at user request. All attachments are maintained in a central repository and controlled through the Configuration Panel.

You can download attachments for any work order. On Android devices, you can only view attachments from the [Download](#) folder in the device. You cannot open the attachment from within the application as you can with an iOS device.

Downloading and uploading attachments are supported for:

- Orders (header level)
- Orders (header level asset)
- Order (object level)
- Operations through Associated Asset
- Operations through PRT
- Notifications (header level)
- Notifications (header level asset)
- Item (object level)
- Item through Associated Asset
- Equipment
- Functional Locations

1.1.1 SAP ERP Transaction Codes for SAP Work Manager

Use the following table as a reference for SAP transaction codes specific to SAP Work Manager functions.

Transaction Codes	Definition
IW21	Create notifications
IW22	Edit notifications

Transaction Codes	Definition
IW23	Display notifications
IW31	Create work orders
IW32	Edit work orders
IW33	Display work orders
CAT2	Record time entries
IE01	Create equipment
IE02	Change equipment
IE03	Display equipment
IL01	Create functional location
IL02	Change functional location
IL03	Display functional location
MM03	Display material
IW41	PM confirmation create
MIGO	Goods issue for work orders
EC86	(Meter Management component) Change disconnect document
EG30	(Meter Management component) Device full replacement
EG31	(Meter Management component) Device full installation
EG32	(Meter Management component) Device full removal
EG33	(Meter Management component) Device technical installation
EG36	(Meter Management component) Device technical removal
EL28	(Meter Management component) Meter reading result single entry

1.2 SAP Work Manager on the SAP Mobile Platform

SAP Work Manager is deployed on the SAP Mobile Platform as an Agentry application. The application is built with the Agentry Toolkit within the SAP EAM and service mobile app SDK.

i Note

For detailed information on the Agentry Toolkit and the SAP EAM and service mobile app SDK, see [Agentry App Development](#).

There are numerous components to the Agentry Toolkit, including the following:

- Runtime environment
- Agentry Editor
- Agentry client
- Agentry test environment (ATE)

- Development server (optional installation)

General overview information on each of these components is provided in this topic. For information the components, including usage, configuration, and other technical details, see the documentation provided with the SAP Mobile Platform and SDK.

Runtime Environment

The runtime environment is a production server system. Within this component, you can define one or more applications of different *archetypes*. An archetype refers to the different types of mobile applications, including the development paradigms, under which mobile applications are developed and deployed. The SAP Work Manager application is developed and deployed under the Agentry archetype. Therefore, it requires the definition of an Agentry application within the runtime environment into which it is deployed. You can modify the application by configuring options within the administration interface, the SAP Mobile Platform Cockpit.

Additionally, you can develop changes to out-of-the-box behavior of the application by using components of the Agentry toolkit within the SAP EAM and service mobile SDK.

SAP Mobile Platform Server

The server is provided within the SAP EAM and service mobile app SDK. It provides the same runtime functionality as the Agentry server component within the SAP mobile runtime environment. It is provided in the Agentry toolkit to allow you to install the server without the need to install the entire runtime environment for development work. The development server is not intended for production use.

Agentry Editor Plug-in to Eclipse

The Agentry Editor plug-in to Eclipse provides a 4GL, point-and-click interface that allows developers to modify the SAP Work Manager application. The Agentry Editor is provided in the SAP EAM and service mobile app SDK as a part of the Agentry toolkit. Agentry applications are stored within the Agentry Editor in the Eclipse workspace as Agentry application projects.

Both the Agentry application project and the Java packages and projects are all managed within a single Eclipse workspace. The developer is presented with a single IDE in which the mobile application as a whole can be maintained and modified. See the document [Agentry App Development](#) in the SAP Mobile Platform documentation for details on working within this toolset.

SAP Work Manager Client

The SAP Work Manager client is provided for each of the client device types supported by the SAP Work Manager application. An installer for the SAP Work Manager client build is provided for Windows operating systems. For devices running iOS, the application is installed from the App store, or from your Apple Enterprise

Server site. For Android devices, the `.apk` file is provided. You can install the `.apk` file from a Web server by navigating to its location within the local network of your environment.

The SAP Work Manager client is an executable run on the client device by the end user. The overall architecture of the Agency toolkit allows for the development of a single application project, which can be deployed on multiple device types. The client processes the business logic developed in the Agency Editor and is deployed to the runtime environment. The client displays the user interface according to the native operating system of the device on which the SAP Work Manager is running.

The Agency Test Environment

The Agency Test Environment (ATE) is among the development tools provided in the Agency Toolkit within the SAP EAM and service mobile app SDK. It is a highly useful tool for developers during the development cycle used for testing the client-side behavior and functionality of your mobile application. It includes numerous debugging and inspection tools to provide insight into the data, action execution, rule evaluation, and other aspects of the behavior of the client at runtime.

The ATE is not an emulator. However, it does possess the capability to mimic the behavior of all the supported client device types. Within the ATE is a full Agency client. Part of the features of the ATE is the ability to select from a list of supported client platforms from within the Agency archetype development paradigm. Once you select a platform, the Agency client tells the SAP Work Manager server that it is a client of that type. The client then receives the user interface components for that platform.

1.3 Setting up the SAP Mobile Platform Server to Support Multiple Languages

Context

English is the default language for SAP Work Manager, so you do not need to perform this procedure if you are using the application in English.

The language pack `.zip` file for SAP Work Manager is named `SAPWorkMgr650LanguagePacks.zip`.

Procedure

1. Unzip all of the translated `.ini` files to the directory where you installed the SAP Mobile Platform server.
2. Edit the `agency.ini` file.
 - a. In the *Configuration* section, add the following line if it does not exist:

```
"localizations="
```

- b. After the equal (=) sign, add the languages used in your application (English is the default language, so you do not need to add it). For example, to add German and Spanish, type

```
localizations=de;es;
```

Note

Only add the required languages necessary for your users:

- ar - Arabic
- cs - Czech
- de - German
- es - Spanish
- fr - French
- he - Hebrew
- hu - Hungarian
- it - Italian
- ja - Japanese
- ko - Korean
- pl - Polish
- pt - Portuguese
- ro - Romanian
- ru - Russian
- sh - Serbian
- sk - Slovakian
- sv - Swedish
- tr - Turkish
- zh-CN - Simplified Chinese

Results

All languages that you defined in the *Localizations* line of the *Configuration* section are supported on the SAP Mobile Platform server.

1.4 Multiple Country Support for Address and Phone Formats

Address Format

You can view, add, or edit addresses based on country specifications. SAP ERP sets up and globally controls the default country format for the application.

SAP Work Manager automatically adapts address layouts to country-specific formats, based on the selected country chosen. Depending on your selection, for instance, you may enable a *State* dropdown or a *Province*

dropdown. For example, Spain uses *Province* while the United States uses *State*. Having a country specification enables your users to select a province from the address dropdown if Spain is the selected country, whereas the dropdown contains states if the United States is the selected country.

Address and Phone Formats List

The supported countries and their printed address formats and phone formats include:

Country	Printed Address Format	Phone Number Format
Argentina (AR)	STREET HOUSE_NO LOCATION POSTL_COD1 CITY REGION COUNTRY	N/A - the phone number is displayed without formatting
Austria (AT)	STREET HOUSE_NO LOCATION POSTL_COD1 CITY COUNTRY	N/A - the phone number is displayed without formatting
Australia (AU)	HOUSE_NO STREET LOCATION CITY REGION POSTL_COD1 COUNTRY	04xx xxx xxx 0x xxxx xxxx
Belgium (BE)	STREET HOUSE_NO LOCATION POSTL_COD1 CITY COUNTRY	N/A - the phone number is displayed without formatting
Brazil (BR)	STREET HOUSE_NO LOCATION POSTL_COD1 CITY REGIO COUNTRY	0 (xx) xxxx-xxxx
Canada (CA)	HOUSE_NO STREET LOCATION CITY REGION POSTL_COD1 COUNTRY	xxx-xxx-xxxx

Country	Printed Address Format	Phone Number Format
Switzerland (CH)	STREET HOUSE_NO	0800 xxx xxx
	LOCATION	0900 xxx xxx
	POSTL_COD1 CITY	0xx xxx xx xx
	COUNTRY	
China (CN)	HOUSE_NO STREET	N/A - the phone number is displayed without formatting
	LOCATION	
	POSTL_COD1 CITY REGION	
	COUNTRY	
Germany (DE)	STREET HOUSE_NO	N/A - the phone number is displayed without formatting
	LOCATION	
	POSTL_COD1 CITY	
	COUNTRY	
Denmark (DK)	STREET HOUSE_NO	xx xx xx xx
	LOCATION	
	POSTL_COD1 CITY	
	COUNTRY	
Spain (ES)	STREET HOUSE_NO	06xx xxx xxx
	LOCATION	0xxx xx xx xx
	POSTL_COD1 CITY REGION	
	COUNTRY	
France (FR)	HOUSE_NO STREET	0 8xx xxx xxx
	LOCATION	0x xx xx xx xx
	POSTL_COD1 CITY	
	COUNTRY	
Greece (GR)	STREET HOUSE_NO	N/A - the phone number is displayed without formatting
	LOCATION	
	POSTL_COD1 CITY	
	COUNTRY	

Country	Printed Address Format	Phone Number Format
Hong Kong (HK)	STREET HOUSE_NO LOCATION POSTL_COD1 CITY REGION COUNTRY	N/A - the phone number is displayed without formatting
Hungary (HU)	POSTL_COD1 CITY STREET HOUSE_NO LOCATION COUNTRY	06 1 xxx-xxxx 06 xx xxx-xxx
Italy (IT)	STREET HOUSE_NO LOCATION POSTL_COD1 CITY REGION COUNTRY	N/A - the phone number is displayed without formatting
Japan (JP)	POSTL_COD1 REGION CITY STREET HOUSE_NO LOCATION COUNTRY	N/A - the phone number is displayed without formatting
South Korea (KR)	POSTAL_COD1 COUNTRY REGION CITY STREET HOUSE_NO LOCATION	N/A - the phone number is displayed without formatting

Country	Printed Address Format	Phone Number Format
Luxembourg (LU)	HOUSE_NO STREET	6x1 xxx xxx
	LOCATION	40 xxxx
	POSTL_COD1 CITY	42 xxxx
	COUNTRY	43 xxxx
		45 xxxx
		47 xxxx
		48 xxxx
		49 xxxx
		4 xxx xx
		xxxx xxxx
	xx xxxx	
Mexico (MX)	STREET HOUSE_NO	N/A - the phone number is displayed without formatting
	LOCATION	
	POSTL_COD1 CITY REGION	
	COUNTRY	
Netherlands (NL)	STREET HOUSE_NO	N/A - the phone number is displayed without formatting
	LOCATION	
	POSTL_COD1 CITY	
	COUNTRY	
Norway (NO)	STREET HOUSE_NO	4xx xx xxx
	LOCATION	9xx xx xxx
	POSTL_COD1 CITY	xx xx xx xx
	COUNTRY	
New Zealand (NZ)	HOUSE_NO STREET	N/A - the phone number is displayed without formatting
	LOCATION	
	REGION	
	CITY POSTL_COD1	
	COUNTRY	
Portugal (PT)	STREET HOUSE_NO	N/A - the phone number is displayed without formatting
	LOCATION	
	POSTL_COD1 CITY	
	COUNTRY	

Country	Printed Address Format	Phone Number Format
Sweden (SE)	STREET HOUSE_NO LOCATION POSTL_COD1 CITY COUNTRY	N/A - the phone number is displayed without formatting
Singapore (AR)	HOUSE_NO STREET LOCATION CITY POSTL_COD1 COUNTRY	N/A - the phone number is displayed without formatting
United Kingdom (UK)	HOUSE_NO STREET LOCATION CITY REGION POSTL_COD1 COUNTRY	See the following table for UK phone number formats
USA (US)	HOUSE_NO STREET LOCATION CITY REGION POSTL_COD1 COUNTRY	(xxx) xxx-xxxx

The following are the phone number formats for the UK:

0800 xxxx	(015395) xxxxx	(01xxx) xxxxxx
0845 46 4x	(015396) xxxxx	(02x) xxxx xxxx
(016977) 2xxx	(016973) xxxxx	03xx xxx xxxx
(016977) 3xxx	(016974) xxxxx	055 xxxx xxxx
(01xxx) xxxxx	(016977) xxxxx	056 xxxx xxxx
0500 xxxxxx	(017683) xxxxx	070 xxxx xxxx
0800 xxxxxx	(017684) xxxxx	076 xxxx xxxx
(013873) xxxxx	(017687) xxxxx	07xxx xxxxxx
(015242) xxxxx	(019467) xxxxx	08xx xxx xxxx
(015394) xxxxx	(011x) xxx xxxx	09xx xxx xxxx
	(01xxx) xxxxxx	

2 Mobile Add-On for ERP Configuration Panel

2.1 Accessing the Mobile Add-On for ERP ConfigPanel

Context

Use the following procedure to access the Configuration Panel, also known as the ConfigPanel.

Procedure

1. Log into SAP.
2. Type the command `/n/syco/configpanel` into the command box and click the green check mark or click *Enter*.

A Web page opens with the Agency SAP Framework Configuration page, or ConfigPanel, for the Mobile Add-On for ERP.

The screenshot displays the 'Agency SAP Framework Configuration' interface. At the top, the title 'Agency SAP Framework Configuration' is visible on the left, and the SAP logo is on the right. Below the title bar, there is a 'Log Off' link. The main content area is divided into several sections:

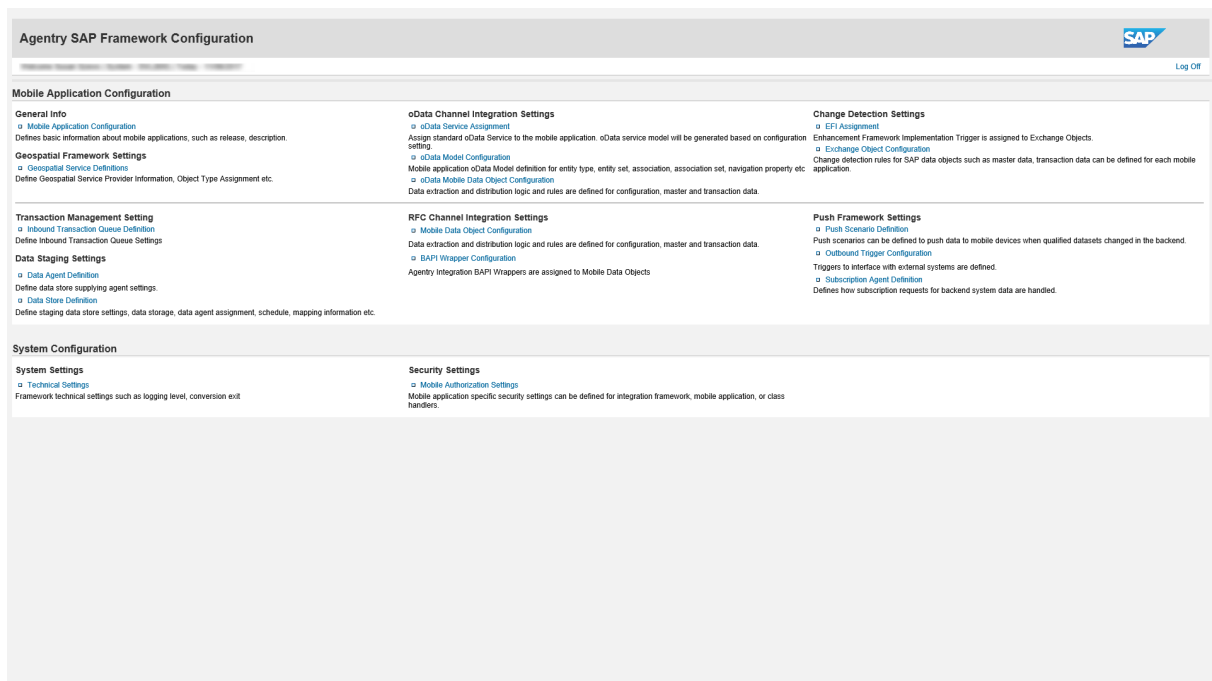
- Mobile Application Configuration**
 - General Info**: Mobile Application Configuration. Defines basic information about mobile applications, such as release, description.
 - Geospatial Framework Settings**: Geospatial Service Definitions. Define Geospatial Service Provider Information, Object Type Assignment etc.
 - Transaction Management Setting**: Inbound Transaction Queue Definition. Define Inbound Transaction Queue Settings.
 - Data Staging Settings**: Data Agent Definition. Define data store supplying agent settings; Data Store Definition. Define staging data store settings, data storage, data agent assignment, schedule, mapping information etc.
 - oData Channel Integration Settings**: oData Service Assignment. Assign standard oData Service to the mobile application. oData service model will be generated based on configuration setting; oData Model Configuration. Mobile application oData Model definition for entity type, entity set, association, association set, navigation property etc; oData Mobile Data Object Configuration. Data extraction and distribution logic and rules are defined for configuration, master and transaction data.
 - Change Detection Settings**: ESI Assignment. Enhancement Framework Implementation Trigger is assigned to Exchange Objects; Exchange Object Configuration. Change detection rules for SAP data objects such as master data, transaction data can be defined for each mobile application.
 - Push Framework Settings**: Push Scenario Definition. Push scenarios can be defined to push data to mobile devices when qualified datasets changed in the backend; Outbound Trigger Configuration. Triggers to interface with external systems are defined; Subscription Agent Definition. Defines how subscription requests for backend system data are handled.
- System Configuration**
 - System Settings**: Technical Settings. Framework technical settings such as logging level, conversion exit.
 - Security Settings**: Mobile Authorization Settings. Mobile application specific security settings can be defined for integration framework, mobile application, or class handlers.

2.2 ConfigPanel Overview

All configuration activities for the Mobile Add-On for ERP are performed through the ConfigPanel.

Changes you make through the ConfigPanel can significantly impact the behavior of the Mobile Add-On for ERP and the SAP Work Manager application. Make and test changes in a development environment first before migrating them to the rest of the SAP system landscape.

The following screenshot shows the home page of the ConfigPanel:



2.2.1 Standard Operations in the Configuration Panel

While each mobile application is unique, certain buttons in the Configuration Panel are standard and are available for all applications.

Filtering by Mobile Application

If more than one mobile application is available on 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 on 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. Information fills in all modifiable fields from the existing item. You can change that information 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 display. After you change the configuration of the item, click [Save](#) to save the changes or [Cancel](#) to discard the changes.

i 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 ConfigPanel - General Info Section

General information settings define and configure how the mobile application, such as SAP Work Manager, functions.

Use the following areas to configure the SAP Work Manager application:

- **Mobile Application Configuration:** Defines basic information about mobile applications, such as application release and description

2.2.2.1 ConfigPanel - Mobile Application Configuration Settings

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
- System Components
- Parameters
- Client Globals
- User Attributes

General Tab

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

The screenshot shows the 'Mobile Application (Display Mode)' configuration page. At the top, there are buttons for 'Create', 'Copy', 'Delete', and 'Change'. Below these are several tabs: 'General' (selected), 'Mobile Status Setting', 'Conversion Exit Setting', 'System Components', 'Parameters', 'Client Globals', and 'User Attributes'. The 'General' tab is active and contains several sections:

- Basic Data:** Mobile Application: SAP_WORK_MANAGER_65, Type: Standard Application, Description: SAP Work Manager 6.5, Release: 6.5.
- User Management Setting:** Disable Automatic User Creation:
- Server Management Setting:** Disable Automatic Server Registration:
- Lifecycle Management:** Application Blocked: Effective Date: [] Time: 00:00:00
- xChange Setting:** Disable Change Detection:
- Inbound Transaction Management:** Inbound Transaction Active: Standard Inbound Queue Id: []
- Multi Backend Setting:** Multi Backend Enabled: System Role: Host
- Administrative Info:** Created By: JEGANESH, Creation Time Stamp: 07/13/2017 23:15:44, Last Changed By: TOPALOVA, Changed Time Stamp: 11/06/2017 21:48:40

- Basic Data fields:
 - **Mobile Application:** For this required field, enter the name of the mobile application, limited to 40 characters.
 - **Description:** For this required field, enter a brief, easy to understand description of the mobile application, limited to 60 characters.
 - **Release:** Release number of the mobile application

- **Type:** Type of application. Note that for SAP Work Manager, the type is *Standard Application*
- User Management Setting:
 - Disable Automatic User Creation:** When checked, a new user GUID is not automatically created when a new mobile client is detected in the system. The system administrator must manually create and maintain mobile users through the Administration portal.
- Server Management Setting:
 - Disable Automatic Server Registration:** When checked, a new server GUID is not automatically created when a new server is detected in the system. The system administrator must manually create and maintain servers through the Administration Portal.
- Life-cycle management fields:
 - **Application Blocked:** If set (checked), the mobile application is disabled. The mobile user can no longer connect to the back-end system for the mobile application. The exchange process is also disabled for the mobile application.
 - **Effective Date and Time:** The date and time this block flag takes effect.
- Inbound Transaction Management fields:
 - **Inbound Transaction Active:** If set (checked), you enable inbound transaction management on the back end.
 - **Standard Inbound Queue ID:** If you are using asynchronous inbound transaction, you can select one queue as the primary queue from the *<Standard Inbound Queue ID>* dropdown. Selecting a primary queue means that all transactions have the same properties and are used for simple mappings. However, out of the box, the selection dropdown is blank. Further customization is needed in the object type assignment of the primary type queue for BAPI wrapper settings, cross references, key mappings, and other settings not detailed in this guide.

When checked, enables a specific mobile application to connect to multiple SAP systems, consisting of one host server and one or more satellite servers.
- Multi-Back End Setting fields:
 - **Multi-Back End Enabled:** 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 listing *Host* or *Satellite*.
 A host system is the connection between SAP and the Agentry application in the SAP Mobile Platform server. The host server provides the logic to the Agentry application and functions as a bridge to the satellite servers. There can only be one host server per system.
 Satellite servers communicate with SAP through the host server.
 To complete multi-back end enabled configuration, configure the host and satellite servers on the *System Components* tab.

Mobile Status Setting Tab

Use the Mobile Status Setting tab to map the available mobile statuses that a mobile data object (MDO) 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 fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application

- **Release:** (Read Only) The release number of the mobile application
- Mobile Status Mapping fields:
 - **Add Status button:** Click [Add Status](#) to create a new mobile status mapping.
 - **Delete Status button:** Click [Delete Status](#) to delete an existing mobile status mapping. To delete a mobile mapping status, click the rectangle to the left of the Object Type column in the row you want to delete and click [Delete](#).
 - **Object Type:** Object type from the Mobile Status Detail section
 - **Mobile Status:** Mobile status from the Mobile Status Detail section
 - **System Status:** System status from the Mobile Status Detail section
 - **User Status:** User status from the Mobile Status Detail section
 - **Initial Status:** When checked, the default status is called after the initial sync of an application based on the specific object type
 - **Skip Status Update:** When checked, the mobile status update is skipped from the back end. Check this flag for each status update to ensure it is a valid status for the application.
- Mobile Status Details fields:
 - **Object Type:** The specific object in a mobile application, ex., [NOTIFICATION](#)
 - **Mobile Status:** Status defined by the mobile application
 - **Label on Mobile:** Not used
 - **System Status:** Standard SAP status code
 - **User Status:** SAP user status code as defined in SAP (note that the status code should be language-independent codes)
 - **Mobile Status Alias List:** Language-specific mobile status aliases can be defined

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 10072 in SAP.

Mobile Application (Display Mode)

Create Copy Delete Change

General **Mobile Status Setting** Conversion Exit Setting System Components Parameters Client Globals User Attributes

Mobile Application Info

Mobile Application: SAP_WORK_MANAGER_65 Release: 6.5
 Mobile App. Desc.: SAP Work Manager 6.5

Mobile Status Mapping

Mobile Status List

Object Type	Mobile Status	Status Attribute 1	Status Attribute 2	System Status	Status Profile	User Status	Initial Status	Skip Update
NOTIFICATION	COMPLETE			I0072			<input type="checkbox"/>	<input type="checkbox"/>
NOTIFICATION	INPROGRESS			I0070			<input type="checkbox"/>	<input type="checkbox"/>
NOTIFICATION	POSTPONE			I0069			<input type="checkbox"/>	<input type="checkbox"/>
NOTIFICATION	STARTED			I0070			<input type="checkbox"/>	<input type="checkbox"/>
NOTIFICATION	SUCCESS			I0157			<input type="checkbox"/>	<input type="checkbox"/>
WORKORDER	COMPLETED			I0045			<input type="checkbox"/>	<input type="checkbox"/>

Mobile Status Detail

Object Type: NOTIFICATION
 Mobile Status: COMPLETE Label On Mobile: COMPLETE
 Status Attribute 1: Status Attribute 2:
 System Status: I0072
 Status Profile: User Status:
 Initial Status: Skip Status Update:

Mobile Status Alias List

*Language	Alias

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.

Conversion Exit Setting Tab

A conversion exit is a function module that converts a value from an external format to an internal, SAP format. A conversion exit can also convert an internal SAP value to a readable, external, value, that is then displayed on a mobile device. However, you may want to completely exclude some conversion routines from your mobile application altogether. Configure those specific conversion exit settings using the *Conversion Exit Setting* tab.

General
Mobile Status Setting
Conversion Exit Setting
System Components
Parameters

Mobile Application Info

Mobile Application: Release:

Mobile App. Desc.:

Conversion Exit Exclusion

Conversion Exit List

	Conversion Exit	Active Flag	Skip Conversion	Skip On Initial	Setting Scope
+	AOBAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	All Conversion Exit
+	EXCRT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	All Conversion Exit
+	TSTPS	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	All Conversion Exit
+	TSTLC	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	All Conversion Exit
+	TSTRG	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	All Conversion Exit

Conversion Exit Detail

Conversion Exit:

Setting Scope: Skip Conversion: Skip On Initial:

Setting Enabled:

- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
- Conversion Exit List fields:
 - **Add button:** Click [Add](#) to create a new conversion exit detail. Fill in the fields in the Conversion Exit Detail section to automatically fill in the fields in this table.
 - **Delete button:** Click [Delete](#) to delete an existing conversion exit detail. To delete a conversion exit detail, click the rectangle to the left of the Conversion Exit column in the row you want to delete and click [Delete](#).
 - **Conversion Exit:** (Read Only) Conversion exit from the Conversion Exit Detail section.
 - **Active Flag (column 1):** (Read Only) When checked, the [Skip Conversion](#) box is checked in the [Conversion Exit Detail](#) section
 - **Skip Conversion (column 2):** (Read Only) When checked, the [Skip on Initial](#) box is checked in the [Conversion Exit Detail](#) section
 - **Skip On Initial (column 3):** (Read Only) When checked, the [Setting Scope](#) box is checked in the [Conversion Exit Detail](#) section
 - **Setting Scope:** (Read Only) Conversion exit scope from the Conversion Exit Detail section

- Conversion Exit Detail fields:
 - **Conversion Exit:** Name of the conversion exit as found in SAP
 - Setting Scope: Choose from the dropdown menu choices:
 - **All Conversion Exit:** Both input and output conversion exit routines are excluded
 - **Input Conversion Exit:** Conversion routines exit when data is sent to SAP
 - **Output Conversion Exit:** Conversion routines exit when data is pulled out of SAP
 - **Skip Conversion:** When checked, the conversion routine is always exited. When you check this box, it automatically checks the box in the first *Active* flag column in the Conversion Exit List table.
 - **Skip on Initial:** When checked, the conversion routine is only excluded when the initial field does not contain a value. If the initial field contains any value, the conversion routine runs. When you check this box, it automatically checks the box in the second *Active* flag column in the table.
 - **Setting Enabled:** When checked, the settings configured for the conversion exit are enabled and the exit is active. When you check this box, it automatically checks the box in the third *Active* flag column in the table.

System Components Tab

Use this tab to define system components in a multi back-end system. Only perform configuration activities in this tab if your application is using a multi back-end system.

Note

Check the *Multi Back-End Enabled* box in the *General* tab of the Mobile Application Configuration pane in order for system component configuration to function.



- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
 - **Multi-back end Enabled:** (Read Only) When checked, the multi back end is activated on the *General* tab.
- System Component List fields:
 - **Add button:** Click *Add* to create a new system component detail. Fill in the fields in the System Component Detail section to automatically fill in the fields in this table.
 - **Delete button:** Click *Delete* to delete an existing system component detail. To delete a system component detail, click the rectangle to the left of the System Component column in the row you wish to delete and click *Delete*.
 - **System Component:** (Read Only) System component from the System Component Detail section
 - **System Role:** (Read Only) System role from the System Component Detail section
 - **RFC Destination:** (Read Only) RFC destination from the System Component Detail section
 - **Active Flag:** (Read Only) When checked, the *Active Flag* box is also checked in the *System Component Detail* section

- **Component Mobile App:** In multi-back end scenarios, when different back end names are used, the component mobile app is the application name that can tie all applications together
- System Component Details fields:
 - **System Component:** For this required field, enter a descriptive name of the component.
 - **RFC Destination:** Define in SAP before configuration in Mobile Add-On for ERP. Use transaction code [SM59](#) in SAP to create or change the RFC destination.
 - **Host:** (Read Only) Identifying host name, defined in SAP
 - **System Number:** (Read Only) Identifying server number, defined in SAP
 - **Client:** (Read Only) Number of the client that the system component connects to, defined in SAP
 - **System Role:** Determines if the system component is a host or a satellite. There can only be one host per multi back-end system.
 - **Active Flag:** When checked, the system component is activated in the multi back-end system
 - **Component Mobile App:** Common application name for multi back-end systems

Parameters Tab

The Parameters tab defines system parameters.

The screenshot shows the 'Mobile Application (Display Mode)' configuration interface. The 'Parameters' tab is active, showing a list of parameters and a detailed view of a selected parameter.

Mobile Application Info

Mobile Application: SAP_WORK_MANAGER_65 Release: 6.5
 Mobile App. Desc.: SAP Work Manager 6.5

Application Parameters

Parameter List

RecNo	Parameter Gro...	Param. Name	Param. Value	Scope	Dep. RecNo	Active	No Change	Comment
000000001	BAPL_CLASS	BSDDocumentCaptureBAPI	com.syclo.sap.component.documentmanagement.ba...	Application	000000000	✓	✓	BAPL_CLASS defines the java class to instantiate f...
000000002	BAPL_CLASS	BSDDocumentLinkPushBAPI	com.syclo.sap.component.documentmanagement.ba...	Application	000000000	✓	✓	BAPL_CLASS defines the java class to instantiate f...
000000003	BAPL_CLASS	BSDDocumentSynchFetchBAPI	com.syclo.sap.component.documentmanagement.syn...	Application	000000000	✓	✓	
000000004	BAPL_CLASS	BillOfMaterialsFetchBAPI	com.syclo.sap.component.equipment.bapi.BillOfMater...	Application	000000000	✓	✓	
000000005	BAPL_CLASS	CharacteristicValueLamCreateBAPI	com.syclo.sap.component.lam.equipment.bapi.Chara...	Application	000000000	✓	✓	
000000006	BAPL_CLASS	CharacteristicValueLamUpdateBAPI	com.syclo.sap.component.lam.equipment.bapi.Chara...	Application	000000000	✓	✓	
000000007	BAPL_CLASS	CharacteristicValueUpdateBAPI	com.syclo.sap.component.equipment.bapi.Characteri...	Application	000000000	✓	✓	
000000008	BAPL_CLASS	ConfirmationCreateBAPI	com.syclo.sap.workmanager.bapi.ConfirmationCreate...	Application	000000000	✓	✓	BAPL_CLASS defines the java class to instantiate f...
000000009	BAPL_CLASS	CrewTimesheetFetchBAPI	com.syclo.sap.component.crewmanagement.timeshe...	Application	000000000	✓	✓	BAPL_CLASS defines the java class to instantiate f...
000000010	BAPL_CLASS	DMSDocumentCaptureBAPI	com.syclo.sap.component.documentmanagement.ba...	Application	000000000	✓	✓	BAPL_CLASS defines the java class to instantiate f...

Parameter Detail

Parameter Group: BAPL_CLASS
 Param. Name: BSDDocumentCaptureBAPI
 Param. Value: com.syclo.sap.component.documentmanagement.bapi.BSDDocumentCaptureBAPI
 Param. Scope: Application
 Rule Id:
 Use Rule:
 Dependent Parameter Id:
 Dependent Parameter Group:
 Dependent Parameter Name:
 Dependent Parameter Value:
 Comment: BAPL_CLASS defines the java class to instantiate for the bapi. Param. Value should be the fully-qualified java class.
 Active Flag:
 No Runtime Change:

- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application

- Parameter List fields:

i Note

The columns in the Parameter List table are read only. Use the Parameter Detail section to make any additions or edits to the table.

- **Add button:** Click [Add](#) to create a new parameter detail. Fill in the fields in the Parameter Detail section to automatically fill in the fields in this table.
- **Delete button:** Click [Delete](#) to delete an existing parameter detail. To delete a parameter detail, click the rectangle to the left of the Parameter Group column in the row you want to delete and click [Delete](#).
- **Parameter Group:** Parameter group from the Parameter Detail section
- **Param. Name:** Parameter name from the Parameter Detail section
- **Param. Value:** Parameter value from the Parameter Detail section
- **Param. Scope:** Parameter scope from the Parameter Detail section
- **Active Flag:** When checked, the [Active Flag](#) box is checked in the [Parameter Detail](#) section
- **No Runtime Change:** When checked, the [No Runtime Change](#) box is checked in the [Parameter Detail](#) section
- **Comment:** Comments from the Parameter Detail section
- Parameter Detail fields:
 - **Parameter Group:** The group to which the parameter belongs. Groups are a means of organizing parameters. References to a parameter include both the group name and the parameter name.
 - **Param. Name:** The unique name of the parameter
 - **Param. Scope:** The scope of the parameter value. There are two options:
 - **Mobile Application:** Value for all users of the application
 - **Mobile User:** Value that can be overridden for individual users. To override a parameter value of a user, see [Parameter Management](#) in the *SAP Mobile Add-On Administrator Guide*.
 - **Param. Value:** The currently configured value of the parameter. References to this parameter return the parameter value
 - **Rule ID:** If enabled, the ID is the rule used at runtime
 - **Use Rule:** When checked, you can define a rule to be used at runtime

i Note

You can either have the parameter active or the rule active, but not both. Checking [Use Rule](#) disables the parameter settings.

- **Rule Input Param:** If the specified rule has optional parameters, define them here
- **Active Flag:** When checked, the parameter is used by the mobile application. Inactive parameters are not available to the mobile application.
- **No Runtime Change:** When checked, the value of the parameter cannot be overridden. The configured value is always the value. If not checked, parameter values can be overridden at runtime through synchronization processing.
- **Comment:** Any comments applicable to the parameter that describe its purpose or value. Comments have no effect on the behavior of the parameter and are provided for reference purposes only.

Client Globals Tab

The Client Globals tab is used to define client globals.

Mobile Application (Display Mode)

General | Mobile Status Setting | Conversion Exit Setting | System Components | Parameters | **Client Globals** | User Attributes

Mobile Application Info

Mobile Application: SAP_WORK_MANAGER_65 Release: 6.5
 Mobile App. Desc.: SAP Work Manager 6.5

Client Global List

RecNo	Global Group	Global Name	Global...	Scope	Dep. RecNo	Active	No Change	Comment
0000000851	HOURS	H22	22	Application	0000000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	HOURS defines static data and is used in constructing data for CTTimelnc...
0000000852	HOURS	H23	23	Application	0000000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	HOURS defines static data and is used in constructing data for CTTimelnc...
0000000853	INBOUND_TRANSACTION_QUEUE	Save		Application	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Reflects "Inbound Transaction Active" flag on the "General" tab.
0000000854	LOGGING	LOG_DIR	agentrylog	Application	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000855	LOGGING	Level	6	Application	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000856	MAIN_MENU_ITEMS	1	WorkOrders	Application	0000000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Main menu item for iPhone
0000000857	MAIN_MENU_ITEMS	10	STOs	Application	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000858	MAIN_MENU_ITEMS	11	Settings	Application	0000000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Main menu item for iPhone
0000000859	MAIN_MENU_ITEMS	2	Notifications	Application	0000000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Main menu item for iPhone
0000000860	MAIN_MENU_ITEMS	3	Crew Man...	Application	0000000000	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Main menu item for iPhone

Client Global Detail

Client Global Group: APPLICATION_CONFIG
 Client Global Name: Attachment.MaxUploadSize Global Scope: Application
 Client Global Value: 500000000
 Rule Id: Use Rule:
 Dependent Client Global Id:
 Dependent Client Global Group:
 Dependent Client Global Name:
 Dependent Client Global Value:
 Comment:
 Active Flag:
 No Runtime Change:

- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
- Client Global List fields:

Note

The columns in the *Client Global List* table are read only. Use the *Client Global Detail* section to make any additions or edits to the table.

- **Add button:** Click *Add* to create a new global. Fill in the fields in the *Client Global Detail* section to automatically fill in the fields in this table.
- **Delete button:** Click *Delete* to delete an existing global. To delete a global, press the rectangle to the left of the *Global Group* column in the row you wish to delete and click *Delete*.
- **Global Group:** Global group from the *Client Global Detail* section
- **Global Name:** Client global group name from the *Client Global Detail* section
- **Global Value:** Client global value from the *Client Global Detail* section
- **Global Scope:** Global scope from the *Client Global Detail* section
- **Active Flag:** When checked, the *Active Flag* box is checked in the *Client Global Detail* section
- **No Runtime Change:** When checked, the *No Runtime Change* box is checked in the *Client Global Detail* section

- **Comment:** Comments from the *Client Global Detail* section
- Client Global Details fields:
 - **Client Global Group:** The group to which the global belongs. Groups are a means of organizing globals. References to a global include both the group name and the global name.
 - **Client Global Name:** The unique name for the global
 - **Global Scope:** The scope of the global value. There are two options:
 - **Mobile Application:** Value for all users of the application
 - **Mobile User:** Value that can be overridden for individual users. To override a parameter value of a user, see [Parameter Management](#) in the *SAP Mobile Add-On Administrator Guide*.
 - **Client Global Value:** The currently configured value of the global. References to the global return this value.
 - **Rule ID:** Name of the ABAP class
 - **Use Rule:** When checked, the rule listed in the *Rule ID* field is active. If this value is active, then the *Client Global Value* field is not used.

i Note

You can either have the global active or the rule active, but not both. Checking *Use Rule* disables the global settings.

- **Rule Input Param:** Parameters to use with the rule. Examples include a key value pair, a user parameter, or a table.
- **Comment:** Displays any comments added to the global to describe its purpose or current value. Comments have no effect on the behavior of the global and are provided for reference purposes only.
- **Active Flag:** When checked, the client global is activated in the system. Inactive globals are not available to the mobile application.
- **No Runtime Change:** When checked, the value of the global cannot be overridden. The configured value in the ConfigPanel is always the value. Globals without this setting can be overridden at runtime through synchronization processing.

User Attributes Tab

Use this tab to create new user attributes.

Mobile Application Info

Mobile Application: Release:
 Mobile App. Desc.:

User Attribute Settings

Attribute List

Attribute Name	Reference Structure Name	Reference Field Name	ValueSet Provider Class
TRUNK_LOCATION	/SMERP/MM_USER_LOCATION_STR		/SMERP/CL_MM_USRATTR_VALUESET

Attribute Detail

* Attribute Name: Description:
 Reference Structure Name: Reference Field Name:
 ValueSet Provider Class:

- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
- User Attribute Settings - Attribute List fields:

You can define any number of user attributes. Add them in the *Attribute Detail* section to populate the attributes in the list. The fields in the Attribute List are read only.

 - **Attribute Name:** (read only) User attribute name.
 - **Reference Structure Name:** (read only) If you assigned this attribute to a reference structure, the name is displayed here.
 - **Reference Field Name:** (read only) If you assigned the attribute to a single field, it is displayed here.
 - **ValueSet Provider Class:** (read only) Each field in the structure can have a lookup table. Valueset associated with this attribute.
- User Attribute Settings - Attribute Detail fields:

Add the attributes here. The only required field is the *Attribute Name*.
 By default, the *Attribute Name* field is a single field, with up to 255 characters. You can define a smaller, single field. For instance, if you want to define a three-character only field, you can define a reference structure and a field. The user attribute can also be a record reference structure.

 - **Attribute Name:** Add a name for the attribute you are defining.
 - **Description:** (Optional) Enter a short description of what this attribute is to be used for.
 - **Reference Structure Name:** (Optional) You can assign the attribute to a structure instead of a single field.
 - **Reference Field Name:** (Optional) You can assign the attribute to a single field.
 - **ValueSet Provider Class:** (Optional) Lookup value table. Each field in the structure can have a lookup table.

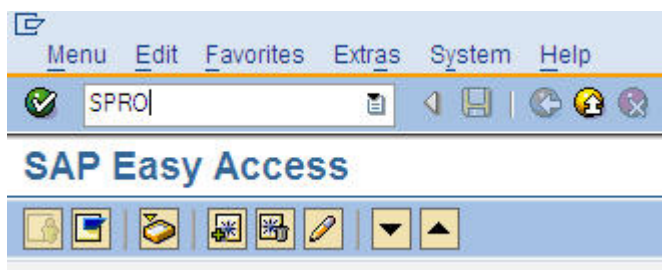
2.3 Mobile Add-On Configuration Panel

Context

Use this procedure to access the Configuration Panel, also known as the ConfigPanel.

Procedure

1. Log in to SAP CRM.
2. Type the command *SPRO* into the command box and click the green check mark or click *Enter*.



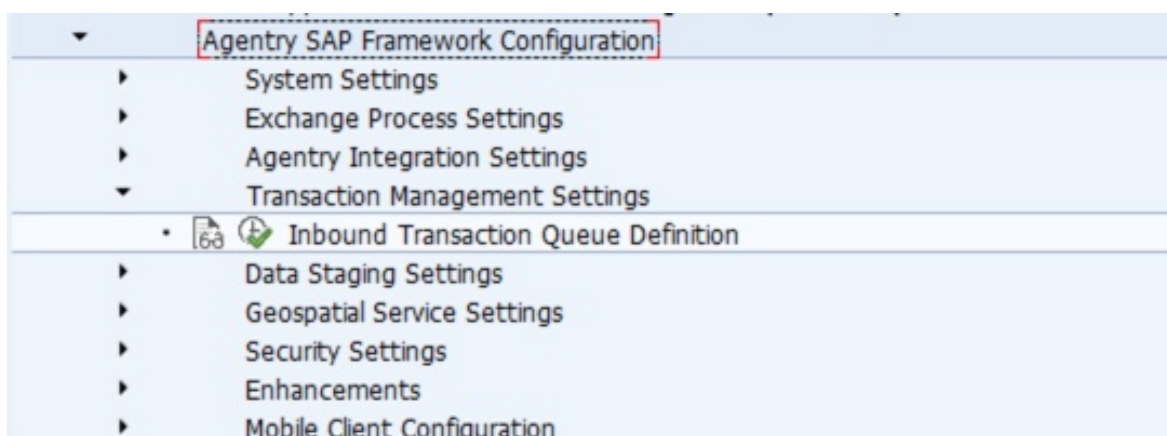
The Customizing: Execute Project screen displays.

3. Click *SAP Reference IMG*.

The Display IMG screen displays.

4. Expand the Mobile Add-On for ERP Configuration line.

Mobile Add-On for ERP submenus display. Expand any one of the submenus to display specific configuration functions and click the clock icon to open the Mobile Add-On for ERP ConfigPanel.



i Note

Click the paper icon to display a screen with a brief description of the specific configuration function.

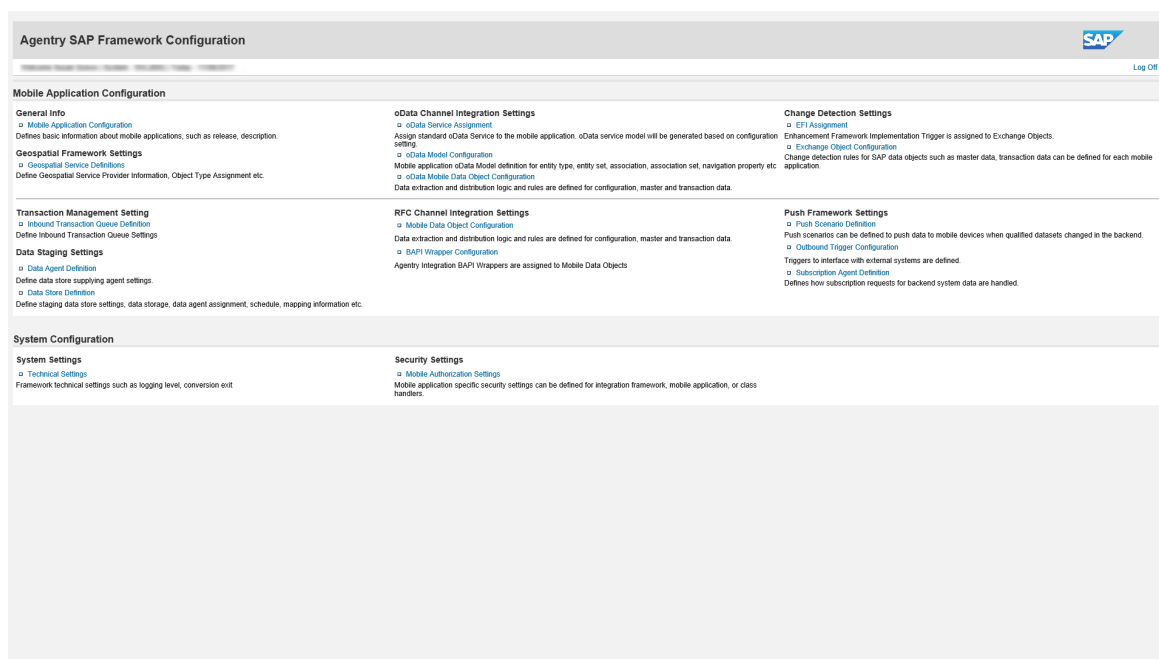
The SAP NetWeaver Web Application Server log on screen opens in a browser window.

- Expand any one of the submenus to display specific configuration functions and click the clock icon to open the Mobile Add-On for ERP Configuration panel.
- Fill in any necessary fields for your specific implementation and click [Log On](#).

The Mobile Add-On for ERP Configuration Panel opens in the browser window. The window that opens corresponds to the submenu line item chosen in SAP.

- Click the ConfigPanel Home link at the top of the screen to navigate to the main configuration page.

The Mobile Add-On for ERP Configuration Panel home page displays.



2.3.1 Configuration Panel Overview

All configuration activities for Mobile Add-On for ERP are performed using the Configuration screen.

Configuration changes made through the Configuration Panel have significant impact to the behavior of the framework component and mobile applications. Make changes in the development environment and fully tested before they are migrated to the rest of the SAP system landscape.

The following user authorizations are required to work with the ConfigPanel:

S_ICF

Authorization object

ICF_FIELD - SERVICE

Authorization field

ICF_VALUE - SYCLOADM

Authorization field

S_TCODE

Authorization object

TCD - /SYCLO/CONFIGPANEL

Authorization field

If you create additional security roles through the Security Settings panel in the ConfigPanel, incorporate them into the system as well.

The screenshot displays the 'Agentry SAP Framework Configuration' interface. At the top right, there is a 'Log Off' button. The main content is organized into several sections:

- Mobile Application Configuration**
 - General Info**: Includes 'Mobile Application Configuration' (Defines basic information about mobile applications, such as release, description) and 'Geospatial Framework Settings' (Includes 'Geospatial Service Definitions' and 'Define Geospatial Service Provider Information, Object Type Assignment etc.').
 - Transaction Management Setting**: Includes 'Inbound Transaction Queue Definition' (Define Inbound Transaction Queue Settings) and 'Data Staging Settings' (Includes 'Data Agent Definition' (Define data store supplying agent settings) and 'Data Store Definition' (Define staging data store settings, data storage, data agent assignment, schedule, mapping information etc.)).
 - oData Channel Integration Settings**: Includes 'oData Service Assignment' (Assign standard oData Service to the mobile application. oData service model will be generated based on configuration setting) and 'oData Model Configuration' (Mobile application oData Model definition for entity type, entity set, association, association set, navigation property etc). It also includes 'oData Mobile Data Object Configuration' (Data extraction and distribution logic and rules are defined for configuration, master and transaction data).
 - oData Mobile Data Object Configuration**: Includes 'RFC Channel Integration Settings' (Includes 'Mobile Data Object Configuration' (Data extraction and distribution logic and rules are defined for configuration, master and transaction data) and 'BAPI Wrapper Configuration' (Agentry Integration BAPI Wrappers are assigned to Mobile Data Objects)).
 - Change Detection Settings**: Includes 'EFI Assignment' (Enhancement Framework Implementation Trigger is assigned to Exchange Objects) and 'Exchange Object Configuration' (Change detection rules for SAP data objects such as master data, transaction data can be defined for each mobile application).
 - Push Framework Settings**: Includes 'Push Scenario Definition' (Push scenarios can be defined to push data to mobile devices when qualified datasets changed in the backend) and 'Outbound Trigger Configuration' (Triggers to interface with external systems are defined). It also includes 'Subscription Agent Definition' (Defines how subscription requests for backend system data are handled).
- System Configuration**
 - System Settings**: Includes 'Technical Settings' (Framework technical settings such as logging level, conversion exit).
 - Security Settings**: Includes 'Mobile Authentication Settings' (Mobile application specific security settings can be defined for integration framework, mobile application, or class handlers).

2.3.1.1 Configuration Panel - Technical Settings

Framework technical settings affect all components of the framework, including mobile data objects, BAPI wrappers, and exchange objects.

2.3.1.2 Configuration Panel - Geospatial Framework Settings

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

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.

SAP Work Manager is delivered with a GIS integration framework. It is delivered out-of-the-box with predefined mobile parameters that are Esri-specific, but you can configure and point to any GIS supplier you choose.

To make the connection to the selected GIS database, properly configure the general back end parameters in the ConfigPanel.

i Note

For full back end customization details, see the following SAP Notes:

- [1982598](#): GIS Integration Support for Mobile Add-On
- [2078078](#): SAP EAM and Service Mobile App SDK - GIS

2.3.1.3 Configuration Panel - Data Staging Framework Settings

By default, data staging is turned off because staged data is stale. The use of data staging is only recommended for more static data.

There are two areas used to configure data staging:

- **Data Store Definition:** Defines staging data store settings, data storage, data agent assignment, schedule, and mapping information
- **Data Agent Definition:** Defines the data store supplying agent settings

2.3.1.4 Configuration Panel - Transaction Management Settings

Transaction management settings are used to define and configure how the mobile application will handle the inbound transaction queue.

There is one area used to configure transaction management:

Inbound Transaction Queue Definition - Defines the basic settings for the inbound transaction queue.

2.3.1.5 Configuration Panel - Mobile Application Settings

Mobile application settings are used to define and configure how the mobile application, such as SAP Work Manager, functions. There are three areas used to configure the mobile application:

- *Mobile Application Configuration* - Defines basic information about mobile applications, such as release and descriptions
- *Push Scenario Definition* - Defines Push Scenarios to push data to mobile devices when qualified data sets change in the back end
- *Subscription Agent Definition* - Defines how subscription requests for back end system data are handled

See the applicable sections in this manual for further information.

2.3.1.6 Configuration Panel - Back End Change Detection Settings

Back end change detection settings are used to define and configure how the mobile application, such as SAP Work Manager, communicates with SAP and the object tables contained within SAP. There are two areas used to configure the back end change detection:

- *Exchange Object Configuration* - Change detection rules for SAP data objects, such as master data and transaction data, defined for each mobile application
- *EFI Assignment* - Enhancement Framework Implementation trigger assigned to Exchange Objects

i Note

You must create tables and objects in SAP and AGENCY before you can create or configure information in the Configuration Panel.

See the applicable sections in this guide for further information.

2.3.1.7 Configuration Panel - Mobile Integration Settings

Mobile integration settings are used to link BAPI wrappers with Mobile Data Objects and encapsulate the business logic related to the mobile application.

There are three areas used to configure mobile integration:

- *Mobile Data Object Configuration* - Data extraction and distribution logic and rules defined for configuration of master data and transaction data
- *BAPI Wrapper Configuration* - AGENCY Integration BAPI Wrappers assigned to Mobile Data Objects
- *Outbound Trigger Configuration* - Triggers to interface with external systems

See the appropriate sections in this manual for further information.

2.3.1.8 Configuration Panel - Security Settings

Use the Security Settings page in the Configuration Panel to set mobile application security parameters at the following levels:

- *System* - Security at this level is application-independent and applies to all components built on the SAP integration framework
- *Product* - Security at this level is at the SAP Work Manager application level
- *Class Handler* - Security at this level is specific to a data object class handler

All security checks are carried out by the SAP Work Manager integration framework at runtime.

2.3.2 ConfigPanel - Technical Settings

The Technical Settings screen in the ConfigPanel is used to change the framework technical settings for application logs viewable in SAP. These application logs are logging levels and conversion exit logs.



The following fields make up the Technical Settings screen:

- **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):** If an SAP object is locked and inaccessible during an update by a mobile device, this parameter controls the number of seconds the underlying component continues to attempt to access the locked object in intervals of one second. If accessing the locked object is still unsuccessful after the wait time, the update process aborts.
- **Internal Conversion Exit Active:** When checked, the framework runtime data manager performs standard SAP external-to-internal format conversion exit for all inbound BAPI parameters. The option is enabled by default. Only an application developer should change the *Internal Conversion Exit Active* setting, as changing it directly impacts the mobile 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. The option is enabled by default. Only an application developer should change the *External Conversion Exit Active* setting, as changing it directly impacts the mobile 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. The check routine sets the SIGN value to *I* and the OPTION value to *EQ* if not specified. The option is enabled by default. Only an application developer should change the *Range Parameter Check Active* setting, as changing it directly impacts the mobile application.
- **Collection Mode:** Determines how system statistic records are written to the database. Two modes are supported currently: Synchronously and Asynchronously. When *Synchronously* is selected, the statistics record is written to the database in real-time during BAPI calls. However, selecting *Synchronously* 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. The statistic collection provides data for the KPI statistics collections found in the Administration Portal. Only an application developer should change the *Statistic Collection Active* setting, as changing it directly impacts the mobile 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.

2.3.3 ConfigPanel - Geospatial Service Definitions Settings

Use the Geospatial Service Definitions screens to define your geospatial service provider and set object type assignments.

i Note

Geospatial Service Definitions are not currently used by the SAP Work Manager application.

2.3.4 Data Store Definition

Use the data store definition screens to define data store settings, data storage, data agent assignment, schedules, and mapping information. In this IMG activity, you define the data staging framework data stores. A data store is a data storage that can provide data to a Mobile Data Object. A data store can consist of multiple data segments, which are related data sets. A data store is populated by assigned data agents and schedules.

For example, you can define a data store and load it with a precalculated data set for a complex table with a large data set for initial download. A defined data store can improve the runtime performance of your mobile client during initial synchronization.

Data Store Definition has the following tabs:

- General Data
- Data Segment
- Agent Assignment
- Field Mappings
- Schedule

General Data Tab

Use the *General Data* tab for basic data information, including the *Data Store ID*, *Store Type*, and *Data Store Description*.

Select the *Reserved for Mobile Data Object ID* field if you want to tie the data store to a single MDO.



Data Segment Tab

Use the *Data Segment* tab to define the Data Supplier Detail information, such as *Data Segment ID*, *Table Name*, and *Segment Alias*.



Agent Assignment Tab

Use the *Agent Assignment* tab to define the Mapping Table Detail, including the *Data Agent ID*, *Data Segment ID*, and the *Agent Parameter Name*.



Field Mappings Tab

Use the *Field Mappings* tab to provide Field Mapping Detail, such as *Table Mapping Assignment*, *Data Agent ID*, *Source Field Name*, *Parameter Name*, and *Target Field Name*.



Schedule Tab

Use the *Schedule* tab to define the Scheduling Detail, such as *Schedule ID*, *Data Agent ID*, *Start Date* and *Start Time*, as well as *End Date* and *End Time*. You can also define a *Frequency* for the schedule as well as an *eCATT Variant Name* and *eCATT Variant Container*.

To execute the loading for the data store, schedule the data store loading utility program /SMFND/CL_DSF_LOADER as an optional background job.



2.3.5 Data Agent Definition

Use this page to define the data store supplying agent settings. Using the *Data Agent Definition* fields, you define a data staging framework data agent. A data agent defines the technical routine that populates a data store. The technical routine definitions include class, function, module, and so on.



2.3.6 Inbound Transaction Queue Definition

Data entries for offline-enabled mobile applications are captured in a local transaction queue on the mobile device. When a connection to the backend SAP system is established, local transactions on the mobile device

are uploaded and posted into the SAP system. From an SAP system perspective, these transactions are known as inbound transactions.

Inbound transaction management is critical to ensure that data updates to the backend SAP system are posted properly and data integrity is preserved.

There are three tabs that make up the inbound queue transaction definition:

- **General Data:** Define your settings for the queue manager handler, dependency checking, and conflict management
- **Object Type Assignment:** Define the technical interface into the queue
- **Security Settings:** Set the *Check Type* for the authorization requirement

The following sections describe the available tabs on the Inbound Transaction Queue Definition page of the ConfigPanel. Use the tabs to define your inbound transaction processing.

General Data Tab

Use this tab to define your settings for the queue manager handler, dependency checking, and conflict management. Take the following in consideration for this tab.

In the *Queue Manager Handler Info* section, select a *Check-in Mode*:

- **Always Check-in:** Always checks the transaction into the queue for processing
- **Dependent Transaction Only:** Checks if there are any dependent transactions already in the queue. If so, the transaction is checked in. If not, the transaction skips the queue and is processed synchronously.
- **Error Processing Only:** Checks the transaction into the queue only if there is an error. Otherwise, the transaction skips the queue and is processed.

Also in the Queue Manager Handler Info section is the *Require User Parameter* checkbox. When checked, two additional fields are displayed: *Parameter ID* and *Activation Token Value*. This parameter selection indicates that the system only activates the inbound transaction if the Parameter ID and Activation Token specified match the profile of the user.

In the *Dependency Checking* section, select a *Dependency Scope*:

- **Cross Application:** Indicates that the transaction can come from any application as long as it is defined properly
- **Application Only:** Indicates that only transactions from a single application are processed.
- **User Only:** Processes transactions from a single user and a single application.

In the *Conflict Management* section, select a *Conflict Mode* to indicate which transactions are processed if there is a conflict::

- **Server Always Win:** When selected, your changes cannot be processed; the server copy is kept intact
- **Last in Win:** When selected, the last update to be made is kept and other changes are overridden
- **Manual:** Does not allow any transactions to be processed until manually done from the back end
- **Not Relevant:** Works the same way as 'Last in Win' but has a slighter faster performance

Object Type Assignment Tab

Use this tab to define the technical interface into the queue. This tab has three subtabs at the bottom of the screen: [Assignment Info](#), [Cross Reference](#), and [Key Mappings](#).

The [Assignment Info](#) subtab allows you to define the following items:

- Request Type
- Request Object
- Request Mobile Data Object - allows you to define what goes into which queue
- Object Type
- Object Group
- Logical System - allows you to define a remote system
- Active - this checkbox indicates that the object type selected is active

The [Cross Reference](#) subtab allows you to define what happens after the transaction is processed (for new service orders or work orders, depending on the application). This subtab works with the Key Mapping subtab to assign key mapping dependencies. When a transaction is processed for a new object and it is successful, the SAP system records a temporary (local) number. Once the assignment is created, the system uses the key mapping to assign the actual number using a cross-reference table in the assignment queue.

Security Settings Tab

Use this tab to set the [Check Type](#) for the authorization requirement. There are four possible check types:

- Checking In
- Checking Out
- Processing
- Review and Approve

2.3.7 ConfigPanel - Mobile Application Configuration Settings

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

Use the Mobile Application Configuration page to set general settings for the entire SAP Work Manager mobile application.

The Mobile Application Configuration screen is comprised of the following tabs:

- General
- Mobile Status Setting
- Conversion Exit Setting
- System Components
- Parameters

- Client Globals
- User Attributes

General Tab

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



- Basic Data fields:
 - **Mobile Application:** For this required field, enter the name of the mobile application, limited to 40 characters.
 - **Description:** For this required field, enter a brief, easy to understand description of the mobile application, limited to 60 characters.
 - **Release:** Release number of the mobile application
 - **Type:** Type of application
- User Management Setting:

Disable Automatic User Creation: When checked, a new user GUID is not automatically created when a new mobile client is detected in the system. The system administrator must manually create and maintain mobile users through the Administration portal.
- Server Management Setting:

Disable Automatic Server Registration: When checked, a new server GUID is not automatically created when a new server is detected in the system. The system administrator must manually create and maintain servers through the Administration Portal.
- Life-cycle management fields:
 - **Application Blocked:** If set (checked), this mobile application is disabled. The mobile user can no longer connect to the back-end system for this mobile application. The xchange process is also disabled for this mobile application.
 - **Effective Date and Time:** The date and time this block flag takes effect.
- Inbound Transaction Management fields:
 - **Inbound Transaction Active:** If set (checked), you enable inbound transaction management on the back end.
 - **Standard Inbound Queue ID:** If you are using asynchronous inbound transaction, you can select one queue as the primary queue from the `<Standard Inbound Queue ID>` dropdown. Selecting a primary queue means that all transactions have the same properties and are used for simple mappings. However, out of the box, the selection dropdown is blank. Further customization is needed in the object type assignment of the primary type queue for BAPI wrapper settings, cross references, key mappings, and other settings not detailed in this guide.
- Multi Back End Setting fields:
 - **Multi Back End Enabled:** 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 listing *Host* or *Satellite*.
A host system is the connection between SAP and the Agency application in the SAP Mobile Platform server. The host server provides the logic to the Agency application and functions as a bridge to the satellite servers. There can only be one host server per system.
Satellite servers communicate with SAP through the host server.

To complete multi back end enabled configuration, configure the host and satellite servers on the [System Components](#) tab.

Mobile Status Setting Tab

Use the Mobile Status Setting tab to map the available mobile statuses that a mobile data object (MDO) 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 fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
- Mobile Status Mapping fields:
 - **Add Status button:** Click [Add Status](#) to create a new mobile status mapping.
 - **Delete Status button:** Click [Delete Status](#) to delete an existing mobile status mapping. To delete a mobile mapping status, click the rectangle to the left of the Object Type column in the row you want to delete and click [Delete](#).
 - **Object Type:** Object type from the Mobile Status Detail section
 - **Mobile Status:** Mobile status from the Mobile Status Detail section
 - **System Status:** System status from the Mobile Status Detail section
 - **User Status:** User status from the Mobile Status Detail section
- Mobile Status Details fields:
 - **Object Type:** The specific object in a mobile application, ex., [NOTIFICATION](#)
 - **Mobile Status:** Status defined by the mobile application
 - **Label on Mobile:** Not used
 - **System Status:** Standard SAP status code
 - **User Status:** SAP user status code as defined in SAP (note that the status code should be language-independent codes)
 - **Mobile Status Alias List:** Language-specific mobile status aliases can be defined

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 10072 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.

Conversion Exit Setting Tab

A conversion exit is a function module that converts a value from an external format to an internal, SAP format. A conversion exit can also convert an internal SAP value to a readable, external, value, that is then displayed on a mobile device. However, you may want to completely exclude some conversion routines from your mobile application altogether. Configure those specific conversion exit settings using the [Conversion Exit Setting](#) tab.



- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
- Conversion Exit List fields:
 - **Add button:** Click [Add](#) to create a new conversion exit detail. Fill in the fields in the Conversion Exit Detail section to automatically fill in the fields in this table.
 - **Delete button:** Click [Delete](#) to delete an existing conversion exit detail. To delete a conversion exit detail, click the rectangle to the left of the Conversion Exit column in the row you want to delete and click [Delete](#).
 - **Conversion Exit:** (Read Only) Conversion exit from the Conversion Exit Detail section.
 - **Active Flag (column 1):** (Read Only) When checked, the Skip Conversion box is checked in the Conversion Exit Detail section
 - **Skip Conversion (column 2):** (Read Only) When checked, the Skip on Initial box is checked in the Conversion Exit Detail section
 - **Skip On Initial (column 3):** (Read Only) When checked, the Setting Scope box is checked in the Conversion Exit Detail section
 - **Setting Scope:** (Read Only) Conversion exit scope from the Conversion Exit Detail section
- Conversion Exit Detail fields:
 - **Conversion Exit:** Name of the conversion exit as found in SAP
 - Setting Scope: Choose from the dropdown menu choices:
 - **All Conversion Exit:** Both input and output conversion exit routines are excluded
 - **Input Conversion Exit:** Conversion routines are exited when data is sent to SAP
 - **Output Conversion Exit:** Conversion routines are exited when data is pulled out of SAP
 - **Skip Conversion:** When checked, the conversion routine is always exited. When you check this box, it automatically checks the box in the first [Active](#) flag column in the table above.
 - **Skip on Initial:** When checked, the conversion routine is only excluded when the initial field does not contain a value. If the initial field contains any value, the conversion routine runs. When you check this box, it automatically checks the box in the second [Active](#) flag column in the table.
 - **Setting Enabled:** When checked, the settings configured for the conversion exit are enabled and the exit is active. When you check this box, it automatically checks the box in the third [Active](#) flag column in the table.

System Components Tab

Use this tab to define system components in a multi back-end system. Configuration in this tab is not necessary if the application does not require a multi back-end system.

i Note

Check the *Multi Back-End Enabled* box in the *General* tab of the Mobile Application Configuration pane in order for system component configuration to function.



- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
 - **Multi back end Enabled:** (Read Only) When checked, the multi back end is activated on the *General* tab.
- System Component List fields:
 - **Add button:** Click *Add* to create a new system component detail. Fill in the fields in the System Component Detail section to automatically fill in the fields in this table.
 - **Delete button:** Click *Delete* to delete an existing system component detail. To delete a system component detail, click the rectangle to the left of the System Component column in the row you wish to delete and click *Delete*.
 - **System Component:** (Read Only) System component from the System Component Detail section
 - **System Role:** (Read Only) System role from the System Component Detail section
 - **RFC Destination:** (Read Only) RFC destination from the System Component Detail section
 - **Active Flag:** (Read Only) When checked, the Active Flag box is also checked in the System Component Detail section
 - **Component Mobile App:** In multi back end scenarios, when different back end names are used, the component mobile app is the application name that can tie all applications together
- System Component Details fields:
 - **System Component:** For this required field, enter a descriptive name of the component.
 - **RFC Destination:** Define in SAP before configuration in Mobile Add-On for ERP. Use transaction code *SM59* in SAP to create or change the RFC destination.
 - **Host:** (Read Only) Identifying host name, defined in SAP
 - **System Number:** (Read Only) Identifying server number, defined in SAP
 - **Client:** (Read Only) Number of the client that the system component connects to, defined in SAP
 - **System Role:** Determines if the system component is a host or a satellite. There can only be one host per multi back-end system.
 - **Active Flag:** When checked, the system component is activated in the multi back-end system
 - **Component Mobile App:** Common application name for multi back-end systems

Parameters Tab

Use this tab to define system parameters.



- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
- Parameter List fields:

i Note

The columns in the Parameter List table are read only. Use the Parameter Detail section to make any additions or edits to the table.

- **Add button:** Click [Add](#) to create a new parameter detail. Fill in the fields in the Parameter Detail section to automatically fill in the fields in this table.
- **Delete button:** Click [Delete](#) to delete an existing parameter detail. To delete a parameter detail, click the rectangle to the left of the Parameter Group column in the row you want to delete and click [Delete](#).
- **Parameter Group:** Parameter group from the Parameter Detail section
- **Param. Name:** Parameter name from the Parameter Detail section
- **Param. Value:** Parameter value from the Parameter Detail section
- **Param. Scope:** Parameter scope from the Parameter Detail section
- **Active Flag:** When checked, the Active Flag box is checked in the Parameter Detail section
- **No Runtime Change:** When checked, the No Runtime Change box is checked in the Parameter Detail section
- **Comment:** Comments from the Parameter Detail section
- Parameter Detail fields:
 - **Parameter Group:** The group to which the parameter belongs. Groups are a means of organizing parameters. References to a parameter include both the group name and the parameter name.
 - **Param. Name:** The unique name of the parameter
 - **Param. Scope:** The scope of the parameter value. There are two options:
 - **Mobile Application:** Value for all users of the application
 - **Mobile User:** Value that can be overridden for individual users. To override a parameter value of a user, see the Administration & Monitoring portal information on parameters.
 - **Param. Value:** The currently configured value of the parameter. References to this parameter return the parameter value
 - **Rule ID:** If enabled, the ID is the rule used at runtime
 - **Use Rule:** When checked, you can define a rule to be used at run time
 - **Rule Input Param:** If the specified rule has optional parameters, define them here
 - **Active Flag:** When checked, the parameter is used by the mobile application. Inactive parameters are not available to the mobile application.
 - **No Runtime Change:** When checked, the value of the parameter cannot be overridden. The configured value is always the value. If not checked, parameter values can be overridden at runtime through synchronization processing.

- **Comment:** Any comments applicable to the parameter that describe its purpose or value. Comments have no effect on the behavior of the parameter and are provided for reference purposes only.

Client Globals Tab

Use this tab to define client globals.



- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application

- Client Global List fields:

i Note

The columns in the *Client Global List* table are read only. Use the *Client Global Detail* section to make any additions or edits to the table.

- **Add button:** Click *Add* to create a new global. Fill in the fields in the *Client Global Detail* section to automatically fill in the fields in this table.
- **Delete button:** Click *Delete* to delete an existing global. To delete a global, press the rectangle to the left of the *Global Group* column in the row you wish to delete and click *Delete*.
- **Global Group:** Global group from the *Client Global Detail* section
- **Global Name:** Client global group name from the *Client Global Detail* section
- **Global Value:** Client global value from the *Client Global Detail* section
- **Global Scope:** Global scope from the *Client Global Detail* section
- **Active Flag:** When checked, the *Active Flag* box is checked in the *Client Global Detail* section
- **No Runtime Change:** When checked, the *No Runtime Change* box is checked in the *Client Global Detail* section
- **Comment:** Comments from the *Client Global Detail* section
- Client Global Details fields:
 - **Client Global Group:** The group to which the global belongs. Groups are a means of organizing globals. References to a global include both the group name and the global name.
 - **Client Global Name:** The unique name for the global
 - **Global Scope:** The scope of the global value. There are two options:
 - **Mobile Application:** Value for all users of the application
 - **Mobile User:** Value that can be overridden for individual users
 - **Client Global Value:** The currently configured value of the global. References to the global return this value.
 - **Rule ID:** Name of the ABAP class
 - **Use Rule:** When checked, the rule listed in the *Rule ID* field is active. If this value is active, then the *Client Global Value* field is not used.

- **Rule Input Param:** Parameters to use with the rule. Examples include a key value pair, a user parameter, or a table.
- **Comment:** Displays any comments added to the global to describe its purpose or current value. Comments have no effect on the behavior of the global and are provided for reference purposes only.
- **Active Flag:** When checked, the client global is activated in the system. Inactive globals are not available to the mobile application.
- **No Runtime Change:** When checked, the value of the global cannot be overridden. The configured value in the ConfigPanel is always the value. Globals without this setting can be overridden at runtime through synchronization processing.

User Attributes Tab

Use this tab to create new user attributes.



- Mobile Application Info fields:
 - **Mobile Application:** (Read Only) The name of the mobile application
 - **Mobile Application Description:** (Read Only) A brief, easy to understand description of the mobile application
 - **Release:** (Read Only) The release number of the mobile application
- User Attribute Settings - Attribute List fields:

You can define any number of user attributes. Add them in the [Attribute Detail](#) section to populate the attributes in the list. The fields in the Attribute List are read only.

 - **Attribute Name:** (read only) This is the user attribute name.
 - **Reference Structure Name:** (read only) If you assigned this attribute to a reference structure, the name is displayed here.
 - **Reference Field Name:** (read only) If you assigned the attribute to a single field, it is displayed here.
 - **ValueSet Provider Class:** (read only) Each field in the structure can have a lookup table. This is the valueset associated with this attribute.
- User Attribute Settings - Attribute Detail fields:

Add the attributes here. The only required field is the [Attribute Name](#).
By default, the [Attribute Name](#) field is a single field, with up to 255 characters. You can define a smaller, single field. For instance, if you want to define a three-character only field, you can define a reference structure and a field. The user attribute can also be a record reference structure.

 - **Attribute Name:** Add a name for the attribute you are defining.
 - **Description:** (Optional) Enter a short description of what this attribute is to be used for.
 - **Reference Structure Name:** (Optional) You can assign the attribute to a structure instead of a single field.
 - **Reference Field Name:** (Optional) You can assign the attribute to a single field.
 - **ValueSet Provider Class:** (Optional) This is the lookup value table. Each field in the structure can have a lookup table.

2.3.8 Push Scenario Definition

Use the Push Scenario Definition link on the ConfigPanel to configure what data is pushed from the SAP back end to the mobile device, and the triggers that initiate a push.

i Note

Push Scenario Definitions are not currently used by the SAP Work Manager application.

2.3.8.1 SAP Cloud Platform, Mobile Service Push Overview

While many on-premise and SAP Cloud Platform, Mobile Service (SCPms) outbound trigger settings are the same, there are some differences. These differences allow the back end to send the push notification to the SCPms.

To send a push notification from SAP to the SCPms, use the following outbound triggers:

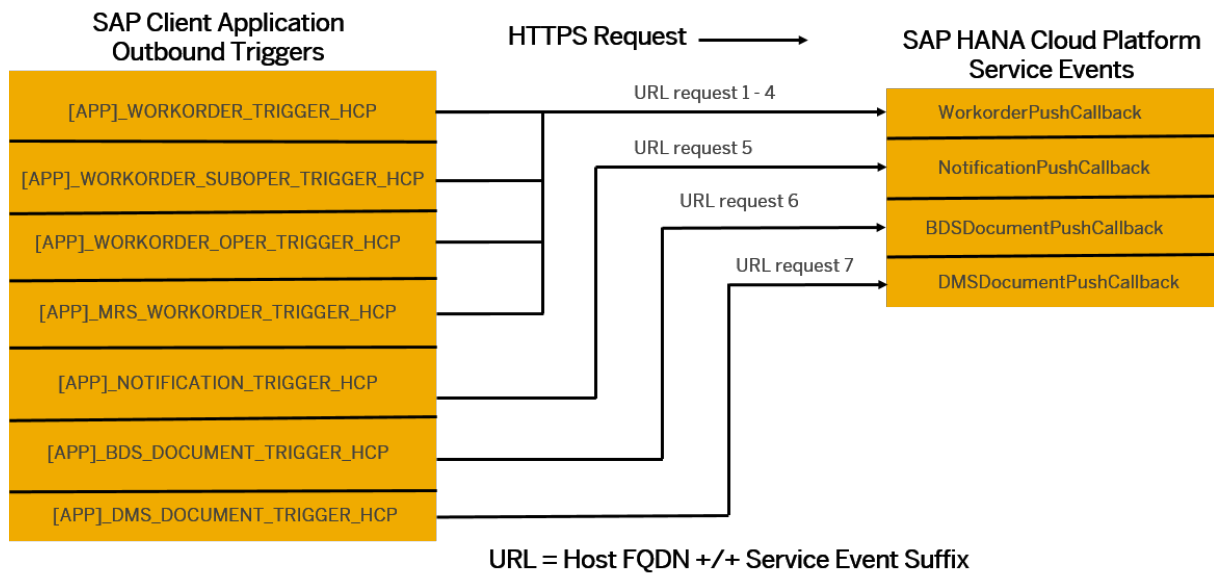
- SWM65_WORKORDER_TRIGGER_HCP
- SWM65_WORKORDER_SUBOPER_TRIGGER_HCP
- SWM65_WORKORDER_OPER_TRIGGER_HCP
- SWM65_MRS_WORKORDER_TRIGGER_HCP
- SWM65_NOTIFICATION_TRIGGER_HCP
- SWM65_BDS_DOCUMENT_TRGGER_HCP
- SWM65_DMS_DOCUMENT_TRIGGER_HCP

In the ConfigPanel, click the [Push Scenario Definition](#) link, [Outbound Trigger](#) tab. Choose your application and the applicable push to configure and click the [Change](#) button.

Each push scenario is assigned to the SAP Mobile Platform (default Active), or SCPms. Change the [Active](#) outbound trigger for the push scenario to the [_HCP](#) outbound trigger and click [Save](#). Repeat this process for each push scenario you use in your application.

- The following are the SCPms service events:
 - WorkorderPushCallback
 - NotificationPushCallback
 - BDSDocumentPushCallback
 - DMSdocumentPushCallback

In the following diagram, you can see how the outbound triggers for the SAP Work Manager application are sent from SAP to the SCPms for push scenarios. Notice especially that the four work order triggers are all sent to the same port ([WorkorderPushCallback](#)). The other three outbound triggers are sent to separate ports for the separate push scenarios, to create a unique URL.



Configuring SCPms on the System Administration and Monitoring Portal

Log into SAP ERP and type the transaction code `/n/syco/admin`. When the Administration and Monitoring Portal displays, click the *Administration* link, and then click *Server Management*. In the *Mobile Application* dropdown field, choose your mobile application and click the *Search* button.

Use the search results if your application is having trouble connecting with the SCPms. Click on a server name and ensure that the *Outbound Trigger URL Type* is set to *Use FQDN when available*. The *Use FQDN when available* is the default setting, but if changed, problems can occur.

2.3.9 Subscription Agent Definition

Subscription Agent Definitions allow you to define how subscription requests for back-end system data are handled.

On Demand subscriptions allow you to define push options. For instance, some accounts, or activities have attachments, which are not automatically pushed down. If subscribed, however, a sales manager can get those attachments on demand, based on the settings configured in the subscription agent.

i Note

Subscription Agent Definitions are not currently used by the SAP Work Manager application.

2.3.10 ConfigPanel - Exchange Object Configuration Settings

The exchange object defines what object in the exchange table to update in the exchange persistent layer, what class handler to call to update the exchange table, and what fields relate to change detection. Use the

Configuration Panel to specify which changes are relevant to your mobile application and what conditions to satisfy to trigger an update action.

The *Exchange Object Configuration* screen consists of 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.

- General data fields:
 - **Exchange Object:** ID of the exchange object, limited to 40 characters
 - **Exchange Object Description:** Brief description of the exchange object, limited to 60 characters
 - **Mobile Application:** Specific mobile application to which the exchange object belongs using a dropdown selection field
 - **Application Area:** Classifies the exchange object based on standard SAP application areas using a dropdown selection field
 - **Reference Business Object:** Standard SAP business object
 - **Exchange Table Name:** Name of the table stored in SAP that contains the technical data
 - **Exchange Table Description:** Brief description of the exchange table
 - **Exchange Lock Object:** SAP lock object used when updating the exchange table
 - **No Exchange Table Update:** When checked, the record is not written to the exchange table in SAP when the record is changed
 - **Days to Keep History:** Number of days the historical data is kept in the exchange table
- Handler Setting field:

Exchange Object Handler: Name of class handler from the repository that is responsible for updating the exchange table
- Collective Run Settings field:

Collective Run Mode: Defines conditions to determine if exchange processing can be executed asynchronously as V3 collective run. The mode options are:

 - **Dynamic:** Collective run mode is determined at runtime by the exchange handler method `DETERMINE_EXEC_MODE`
 - **Not Allowed:** Not allowed to switch to collective run mode
 - **Activated:** Always execute asynchronously in v3 collective run
 - **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 fields:
 - **Active Flag:** When checked the exchange object is in an active state. If unchecked, the exchange object performs no actions.
 - **Use in Linkage Processing Only:** If checked, the exchange object is only allowed during linkage processing. If not checked, the original EFI triggered exchange processing takes place.

Example:

The following screen sample shows that the exchange process is enabled for SWM65_RESERVATION. Any changes to Reservation master data are recorded in the exchange table and transmitted to the client during the next transmit.



Change Detection Field Selection Tab

The *Change Detection Field Selection* tab allows you to optimize the change detection process for mobile applications. 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 has been made. If the *Active* flag is not checked for a field, any value changes made to that field are not detected or recorded to SAP during the exchange process. By default, all fields are initially checked.



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

- Exchange Object Info fields:
 - **Exchange Object:** A noneditable field that contains the ID of the exchange object.
 - **Exchange Object Description:** A noneditable field that contains a brief description of the exchange object.
 - **Exchange Object Handler:** A noneditable field that contains the name of class handler from the repository that is responsible for updating the exchange table.
- Exchange Object Field Selector fields:
 - **Field Catalog:** All fields that can be detected by the class handler when changes are made, grouped by the technical table name of the SAP business object. The field catalog is a noneditable field.
 - **Active Flag:** When checked, either the table or a field within a table is active. Any value change to the selected field is detected by the class handler.

i Note

Checking the *Active Flag* box on a Table row selects all fields within the table.

- **Short Description:** Brief description of the table or the field within the table. The short description is a noneditable field.
- Selection Proposal options:

In a typical installation, it is not desirable to have all fields in all exchange tables checked as active for change detection. Rather, only the fields that are active on the mobile data object that are brought down to the mobile device should also be active in the exchange object.

Based on mobile data object usage in the mobile application, the Selection Proposal examines the active flags that are checked for the table fields of an exchange object. The Selection Proposal then provides recommendations to the administrator on which fields in the exchange object to check or uncheck.

Change Detection Condition Filter Tab

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



- Exchange Object Info fields:
 - **Exchange Object:** A noneditable field that contains the ID of the exchange object.
 - **Exchange Object Description:** A noneditable field that contains a brief description of the exchange object.
 - **Exchange Object Handler:** A noneditable field that contains the name of class handler from the repository that is responsible for updating the exchange table.
- Exception Settings fields:
 - **Ignore Data Creation:** When checked, new records, data, or both are created are not processed to the exchange table
 - **Ignore Data Deletion:** When checked, deleted records, data, or both are not processed to the exchange table
 - **Ignore Data Update:** When checked, updated records, data, or both are not processed to the exchange table
- Defined Filters list:

The Defined Filters box lists all data filters supported by the class handlers.
- Rule Editor fields:
 - **Filter Name:** Name of the filter as defined in the class handler method. This information is defined by the class handler developer and is not editable.
 - **Reference Table Name:** Technical name of the SAP database table where this filter is applied. This information is defined by the class handler developer and is not editable.
 - **Reference Field Name:** Technical name of the SAP database table field where this filter is applied. This information is defined by the class handler developer and is not editable.
 - **Data Filter Rule Key:** Internal technical key used by the framework at runtime
 - **Range Value:** Use the following values to set the range:
 - **Sign:** Value for the SAP range table column SIGN
 - **Option:** Value for the SAP range table column OPTION
 - **Low Value:** Value for the SAP range table column LOW
 - **High Value:** Value for the SAP range table column HIGH
 - **Active Flag:** When checked, the rule is active
- Rule List table:

The Rule List table displays a list of rules that have been defined.

 - **Rule No.:** Number of the rule that is defined, in chronological order
 - **Rule Type:** Rule type, automatically assigned by the rule type selected in the DOF Rule Type field
 - **Rule Value:** Internal rule value saved by SAP
 - **Active Flag:** When checked, the rule is active

Data Segment Settings Tab

The Data Segment Settings tab defines data segment requirement details for the specific exchange object. Data segment setting is optional for any given exchange object. If no data segment setting is defined, an exchange object is always called by the EFI event trigger if the exchange object is assigned to the EFI in the [EFI Assignment Configuration](#) settings.

By defining data segment settings, additional criteria is applied when determining whether an exchange object is called by an EFI event trigger. Even when an exchange object is assigned to the EFI in the EFI Assignment configuration settings, the exchange object is only called by the EFI event trigger when the data segment requirements are met.



Linkage Settings

The Linkage Settings tab allows the exchange objects that are linked together to communicate with each other. The communication is one-directional, with the exchange object sending information to the object(s) listed in the Linked Exchange Objects List. When there is a value change to the exchange object, that value change information is passed on to the linked exchange objects. The linked exchange objects then go through additional processes related to the value change.



- Exchange Object Info fields:
 - **Exchange Object:** A noneditable field that contains the ID of the exchange object.
 - **Exchange Object Description:** A noneditable field that contains a brief description of the exchange object.
 - **Exchange Object Handler:** A noneditable field that contains the name of class handler from the repository that is responsible for updating the exchange table.
- Linkage Settings field:

With the Linkage Hierarchy, you can go 'n' levels deep with linked objects. Any node changes triggers changes to the lower-level nodes linked to the parent node. These relationships are defined in the [Linked Exchange Objects](#) list.

For example, the exchange object `<APP>_MEASURING_POINT` is linked to the following exchange objects:

 - `<APP>_FLEET`
 - `<APP>_EQUIPMENT`
 - `<APP>_FUNC_LOCATION`

If the Measuring Point data changes, then the linked exchange objects also change.

Maximum Linkage Hierarchy Level:
- Linked Exchange Objects list:
 - **Add Linkage button:** Use this button to add a new linked exchange object. Click [Add Linkage](#) and use the fields in the Linkage Detail section to add information.
 - **Delete Linkage button:** Use this button to delete a linkage. To highlight the row you want to delete, click the rectangle to the left of the Target Exchange Object cell and click [Delete Linkage](#).

- **Target Exchange Object:** Displays the target exchange object selected in the Linkage Detail section.
- **Linkage Type:** Displays either an *A* for asynchronous or an *S* for synchronous, selected in the [Linkage Detail](#) section.
- **Active Flag:** When checked in the [Linkage Detail](#) section, the linkage between exchange objects is active.
- Linkage Detail fields:
 - **Target Exchange Object:** The exchange objects that are linked to the exchange object listed in the [Exchange Object Info](#) section.
 - **Linkage Type:** Currently, Synchronous is the only option available. When a value change occurs to the exchange object, notification to the linked exchange object is performed in real-time.
 - **Exclude Data Creation / Update / Deletion:** The linkage for a source exchange object to a target exchange may be limited by the action done on the source object. The possible actions are Create, Update, and Delete. By checking any of these three 'exclude' boxes, the linkage is not triggered for that action.
 - **Active Flag:** When checked, the linkage between the exchange object and the target exchange object is active.

Push Settings Tab

If the exchange object is part of a push instance, you must configure the object in the [Push Settings](#) tab before the object can appear in the Source Object dropdown menu of the push scenario definition.



- Exchange Object Info fields:
 - **Exchange Object:** A noneditable field that contains the ID of the exchange object.
 - **Exchange Object Description:** A noneditable field that contains a brief description of the exchange object.
 - **Exchange Object Handler:** A noneditable field that contains the name of class handler from the repository that is responsible for updating the exchange table.
- Push Settings field:
 - **Push Relevant:** When checked, the exchange object is listed as a selection in the [Source Object](#) dropdown list in the [Push Scenario Definition](#) screen.

2.3.11 ConfigPanel - EFI Assignment Settings

Enhancement Framework Implementation (EFI) source code plug-ins are implemented by the Mobile Add-On for ERP for each business object where you are implementing 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 has been created, modified, or deleted. The EFI then hands those images to the exchange object, which continues with the data processing. Therefore, you link the EFIs to the appropriate exchange objects.

Enhancement Implementation Includes

The Enhancement Implementation Includes list is a tree of the include file list in the package. Click the arrow to the left of the first item to expand the list:



General Tab

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



- **EFI Type:** Select one of the following options for EFI type:
 - **Standard EFI Include:** Traditional way to implement EFI and configure the EFI assignments
 - **EFI Event Handler** Implement EFI using an ABAP class-based approach. If you select *EFI Event Handler*, you develop EFI implementation as a subclass of `/SMFND/CL_CORE_EFI_EVENT_BASE`. Available EFI event handler classes are displayed in the dropdown field. The EFI event handler approach provides a more robust functionality and is recommended for new EFI implementations.
- **EFI Include Name:** Source code plug-in file name
- **Description:** Short description of the EFI. The field is automatically filled in when you select the EFI Include Name and is not editable.
- **Package:** Package where the EFI is located. This field is automatically filled in when you select the EFI Include Name and is not editable.

Assignment Tab

Use the Assignment tab to modify the EFI assignments.



- **EFI Info fields:**
The general EFI information fields in this section are taken from information in the General tab and are not editable.
- **EFI Assignment List:**
The EFI Assignment table displays which plug-ins are assigned to a specific include file. All column information is replicated in the Assignment Detail section directly below the table.
To highlight an individual row, click the gray square to the left of the *EFI Include Name* column in that row.
- **Assignment Detail fields:**
Information in this section changes depending on which row is highlighted in the *EFI Assignment List* section table.
 - **Mobile Application:** The specific mobile application and its release number. This field is noneditable.

- **Exchange Object:** Exchange object to which the EFI include file is assigned.
- **Exch. Object Desc:** A brief, easy to understand description of the exchange object, limited to 60 characters.
- **Active Flag:** When checked, the exchange object is in an active state. If unchecked, the EFI is not linked to the assigned mobile data object

2.3.12 Mobile Data Object Configuration

A mobile data object, or MDO, represents a mobile semantic view of data and activity combination for an SAP business object.

Mobile data objects are data repositories in the namespace that can get, create, update, and delete information in SAP. They encapsulate the business logic of mobile applications by defining transactions, data structures, and business rules.

There are three types of MDOs:

- **Data Table (DT):** A simple representation of SAP BusinessObjects KEY and VALUE.
- **Complex Table (CT):** A two-dimensional representation of a business object with a single table of multiple columns.
- **Standard Data Object (DO):** A multi-dimensional representation of a business object with multiple tables representing different subsets of the business object.

⚠ Caution

Only administrators or mobile application developers should create, copy, or change mobile data objects.

Use the ConfigPanel to modify MDO properties such as object type, class handlers, data filters, and other settings. For example, instead of modifying BAPIs to change what information is retrieved from SAP and pushed out to mobile devices, administrators use the ConfigPanel to modify MDOs and set up data filter rules.

The Mobile Data Object Configuration screen has the following tabs:

- General Setting
- ResultSet Field Selection
- Data Filter
- Data Staging
- Proxy Setting

General Setting Tab

Use the *General Setting* tab to modify the general settings for a chosen mobile data object.



- **Basic Data:** The Basic Data section provides general information about the specific mobile data object. The basic data information is used in tabs of the Mobile Data Object screen.

- **Mobile Data Object ID:** Required field. Enter the name of the MDO, limited to 40 characters
- **Description:** Required field. Enter a brief, easy to understand description of the MDO, limited to 60 characters
- **Data Object Type:** A dropdown list of the three MDO types:
 - **Data Table:** A simple representation of the SAP BusinessObjects KEY and VALUE
 - **Complex Table:** A two-dimensional representation of the business object with a single table of multiple columns
 - **Standard Data Object:** A multi-dimensional representation of a complete business object. It can have multiple tables representing different subsets of the SAP BusinessObjects.
- **Mobile Application:** Name of the mobile application. Choose the mobile application from the dropdown list.
- **Reference Business Object:** SAP business object for which the MDO is being created.
- **Data Object Handler Settings:** Use the Data Object Handler Settings section to configure the methods of the MDO.
 - **Data Object Handler:** Name of the ABAP OO class handler from the class repository of SAP. The ABAP OO class handler is developed by the application developer with predefined business logic and scope to perform fetch, create, delete, or update activities for an SAP business object.
 - **Get Method:** Method defined in the class handler that fetches data for the underlying SAP business object (optional)
 - **Create Method:** Method defined in the class handler that creates data for the underlying SAP business object (optional)
 - **Update Method:** Method defined in the class handler that updates data for the underlying SAP business object (optional)
 - **Delete Method:** Method defined in the class handler that deletes data for the underlying SAP business object (optional)
 - **Skip Exception Processing:** When checked, the exception processing step is not invoked if an exception occurs during MDO handling
- **Exchange Object Settings:** Allows you to associate the MDO with an exchange object. Configure the exchange object using the [Exchange Object Configuration](#) screen in the ConfigPanel.
 - **Exchange Object:** Name of the exchange object as defined in the SAP mobile exchange persistent layer. Specify an exchange object from the dropdown menu if the selected class handler uses the mobile exchange persistent layer to determine data exchanges to the mobile application.
 - **Enable Conv. Exit Overwrite:** When checked, you can define specific internal and external conversion settings in the [Technical Settings](#) screen of the ConfigPanel.
- **Localization Settings:** When you enable localization in the MDO configuration, you can assign a localization language to an MDO, which is used for language related distribution. For example, a complex table MDO with localization language enabled returns language relevant descriptions in the specified localization language instead of the user logon language.
 - **Enable Localization Setting:** Check to allow localization
 - **Language:** Assign the localization language to the MDO
- **Middleware Reference Info:** If the middleware is specified, the MDO performs the standard exchange process as well as a lookup in the client object register table to determine what information the client contains. If data was removed from the client that still exists in the SAP table, the data is readded to the client during the transmit.
 - **Reference Middleware Object Type:** Middleware objects are set through the Administration portal (optional)
- **Activation:** Use the checkbox to enable or disable an MDO in the application without deleting the MDO.

- **Data Object Active:** When checked, the MDO is in an active state. If unchecked, the MDO performs no actions

ResultSet Field Selection Tab

When you enable a field selector function for a class handler, you can then select fields for the GET method to populate. The class handler is mobile application neutral. It supplies more data than the mobile application needs. Therefore, to preserve system performance, customize field usage settings to only retrieve required data for the mobile application. The ability to customize prevents the need to develop a new class handler for each mobile application.



- **Handler Info fields:**
 - **Mobile Data Object ID:** Name of the MDO
 - **Description:** Description of the mobile data object, limited to 60 characters. Required field.
 - **Data Object Handler:** Name of the ABAP OO class handler from the class repository of SAP. The ABAP OO class handler is developed by the application developer with predefined business logic and scope to perform fetch, create, delete, or update activities for an SAP business object.
 - **Get Method:** Name of the GET method set in the *General Setting* tab.
- **Field Selection Detail fields:**
 - **Field Catalog column:** Lists all the fields that can be returned by the class handler method, grouped in the order of the name of the class handler method, SAP table name, and field name. To display all information in this column, click the arrows to the left of a name to expand the row.
 - **Field Active column:** When checked, the data for the selected field is returned by the class handler method
 - **Field Description column:** Description of the specific field in the SAP table
 - **Data Format column:** How data is presented
 - **Sort Options:** When a large amount of information is presented, use the sort options to find the information required easily. When you select a different radio button, the rows collapse. Expand them again to display the new field sorting.

Data Filter Tab

When a data filter function is enabled for a class handler, the option exists to define various types of filter rules to control what data a user can view on the mobile application based on the business process of the customer. In an SAP environment, each user is assigned a role-based profile with authorization restrictions for what data is viewed and which activities performed.

For example, a user who works for a specific plant should not be able to view data for another plant. Data filter rules allow you to restrict data access for mobile applications. Data filters can be user-dependent or applied to the entire mobile application.



- **Handler Info fields:**
 - **Mobile Data Object ID:** Name of the MDO
 - **Description:** Description of the mobile data object, limited to 60 characters. Required field.
 - **Data Object Handler:** Name of the ABAP OO class handler from the class repository of SAP. The ABAP OO class handler is developed by the application developer with predefined business logic and scope to perform fetch, create, delete, or update activities for an SAP business object.
- **Defined Filters list:**

The Defined Filters tree lists all data filters supported by the class handlers defined in the *Data Object Handler Settings* field in the *General Setting* tab. To expand the tree, click the arrows to the left of the class handler methods to display the filters associated with the methods.
- **Rule Editor fields:**
 - **Method Name:** Name of the class handler method where the data filter is defined. The data filter function is only supported for the GET method
 - **Filter Name:** Name of the filter as defined in the class handler method. The filter name is defined by the class handler developer and is not editable.
 - **Reference Table Name:** Technical name of the SAP database table where this filter is applied. The reference table name is defined by the class handler developer and is not editable.
 - **Reference Field Name:** Technical name of the SAP database table field where this filter is applied. The reference field name is defined by the class handler developer and is not editable.
 - **Data Filter Rule Key:** Internal technical key used by the framework at runtime
 - **DOF Rule Type:** Type of rule

Each rule set has different settings options:

 - **User Profile Parameter rule**
 - **Parameter ID:** Memory parameter ID as defined in SAP and specified in the user profile. Click the icon to the right of the field box to perform a search on all available parameter IDs.
 - **Description:** Description of the memory parameter ID. The description field is not editable and is automatically filled in when the parameter ID is selected.
 - **Active Flag:** When checked, the rule is active
 - **Static Value in Range Format rule**
 - *Sign*, *Option*, *Low Value*, and *High Value* are the values for the applicable columns in the *SAP Range Table*.
 - **Syclo Filter Class Handler rule**
 - **Syclo Data Filter Handler:** Name of the handler class as defined in the system
 - **Parameter:** Additional processing information that is passed to the class handler. The parameters are entered as free text, and the syntax of the parameter stream is defined by the developer.
 - **Active Flag:** When checked, the rule is active
 - **Runtime Session Data rule**
 - **Runtime Session Data Name:** Session ID within the specified group
 - **Runtime Session Data Group:** Name of the session group
 - **Active Flag:** When checked, the rule is active
- **Rule List table:** The Rule List table displays a list of rules that have been defined
 - **Rule No.:** Number of the rule that is defined, in chronological order
 - **Rule Type:** Rule type, automatically assigned by the rule type selected in the *DOF Rule Type* field
 - **Rule Value:** Internal rule value saved by SAP

Data Staging Tab

If an application processes a large amount of objects, data staging of the objects can assist with processing times. If an object is configured for data staging, the data within the object is stored as a package and is split into packets. The data can contain metadata and tagging for easy life-cycle management and data lookup. Standard APIs are provided for package management.



- **Handler Info fields:**
 - **Mobile Data Object ID:** Name of the MDO
 - **Description:** Description of the mobile data object, limited to 60 characters. Required field.
 - **Data Object Handler:** Name of the ABAP OO class handler from the class repository of SAP. The ABAP OO class handler is developed by the application developer with predefined business logic and scope to perform fetch, create, delete, or update activities for an SAP business object.
- **Data Staging Setting fields:**
 - **Get, Create, Update, and Delete Method Setting:** When a checkbox is marked for a specific method, data staging is active for that method

Proxy Setting Tab

Use the [Proxy Setting](#) tab to define any proxy settings.



- **Handler Info fields:**
 - **Mobile Data Object ID:** Name of the MDO
 - **Description:** Description of the mobile data object, limited to 60 characters. Required field.
 - **Data Object Handler:** Name of the ABAP OO class handler from the class repository of SAP. The ABAP OO class handler is developed by the application developer with predefined business logic and scope to perform fetch, create, delete, or update activities for an SAP business object.
- **Proxy Settings fields:** The following fields are available for each of the methods:
 - **System Component:** Identifies the remote system
 - **Proxy Type:** BAPI Proxy is the only supported proxy type
 - **Proxy Name:** BAPI name of the remote system
 - **Proxy Active:** When checked, proxy settings are active for the specific method

2.3.13 BAPI Wrapper Configuration

A BAPI wrapper is created by the application developer to expose SAP data and business logic to the mobile application. By design, the BAPI wrapper does not contain any business logic. Each BAPI wrapper is assigned to a specific method type (GET, CREATE, UPDATE, or DELETE) of an MDO to perform the required business

logic. By decoupling the business logic from the BAPI wrapper, it is possible to switch MDOs without affecting the underlying mobile application definition.

The *BAPI wrapper Configuration* screen contains the following tabs:

- General
- Assignment

General Tab

Use the *General* tab to modify general settings for a selected BAPI wrapper.



- **BAPI Wrapper Info fields:**
 - **BAPI Wrapper Name:** Technical name of the Remote Function Call (RFC) function module defined in the system
 - **Description:** Description of the BAPI wrapper, limited to 60 characters (required)
 - **Function Group:** Function group to which the BAPI belongs
 - **Function Group Description:** Description of the function group, limited to 60 characters
 - **Package:** SAP group to which the function group and the BAPI wrapper both belong
- **Technical Info:** The Technical Info, or BAPI wrapper list, provides an expandable tree of the available BAPI wrapper function groups and the BAPI wrapper names associated with each function group. Use the arrows to the left of the function name to display all BAPI wrapper names under the function group.

Assignment Tab

Use the Assignment tab to hook up BAPI wrapper assignments to mobile data objects. In this tab, you can change the assignment of a BAPI wrapper to specific mobile data objects, assign a new mobile data object to the BAPI wrapper, or delete a mobile data object assignment from the BAPI wrapper.



- **BAPI Wrapper Info fields:**
 - **BAPI Wrapper Name:** Technical name of the Remote Function Call (RFC) function module defined in the system
 - **Description:** Description of the BAPI wrapper, limited to 60 characters (required)
- **Mobile Data Object Assignment List:**
 - **Mobile Data Object Assignment table:** Table that displays which mobile data objects are assigned to each BAPI wrapper. All column information is replicated in the *Assignment Detail* section below the table.
To highlight an individual row, click the gray square to the left of the *Mobile Application* column in that row.
- **Assignment Detail fields:** Information in these fields changes depending on which row you highlight in the *Mobile Data Object Assignment List* table

- **Mobile Application:** The specific mobile application and its release number. This field is noneditable
- **Mobile Data Object ID:** The name of the mobile data object in a dropdown field
- **Description:** A brief, easy to understand description of the mobile data object, limited to 60 characters. This field is noneditable
- **Method Type:** Mobile data object method (*GET*, *CREATE*, *UPDATE*, or *DELETE*) that is assigned to the BAPI wrapper
- **Active Flag:** When checked, the mobile data object is in an active state. If unchecked, the mobile data object performs no actions
- **Default Assignment:** When checked, the specific mobile data object assigned to that BAPI wrapper is primary. If no mobile data object ID is specified in the standard BAPI wrapper input parameter `<IS_BAPI_INPUT-DO_ID>`, then the primary mobile data object is used during runtime. It is possible to assign multiple MDOs to the same BAPI in a single mobile application. To override the default MDO assignment at runtime and to specify the desired MDO, define the `<DO_ID>` field in `<IS_BAPI_INPUT>`.

2.3.14 Outbound Trigger Configuration

An outbound trigger allows a mobile application to interface with external systems such as Agency middleware server from SAP. You can integrate outbound triggers into standard mobile application processes, such as push processing. You can define different types of outbound triggers, including HTTP triggers, file triggers, and web service triggers.

i Note

Outbound trigger configuration is not currently used by the SAP Work Manager application.

2.3.15 ConfigPanel - Security Settings

Security settings are used to provide additional rules and roles on top of the standard SAP-provided rules and roles.

For detailed information on required and essential authorizations, and procedures on enabling mobile-specific authorization checks, see the [SAP Work Manager Security Guide](#).

2.3.15.1 Security Settings - System Security

Use this tab to configure system security settings that are mobile application-independent. System security settings apply to all applications running on the framework.

Security Settings - System Security

- *Authorization Field Name*: This read-only field displays the standard SAP authorization object name.
- *Authorization Field Value*: Free-text field. Text entered depends on developer implementation in SAP. This is a non-editable field.
- *System Admin Indicator*: When Rule Type: User Role is selected, taken from the System Admin Indicator field in the Rule Detail section. This is a non-editable field.

Rule Detail - Rule Type: User Role

- *Rule Type: User Role*: In addition to the standard SAP user profile rules, the user role can add restrictions on what a user can or cannot see in menus or other mobile application screens.
- *Role*: User role in SAP. To search for a user role, click on the white box icon to the right of the Role field to display the Role Selection search window.
- *Name*: Brief description of the role. This is a non-editable field.
- System Admin Indicator: Drop-down menu with four choices:
 - System administrator: User role can view system activity and make changes to system administration setup in the Administration portal.
 - System administration - View only: User role can view the system activity in the Administration portal, but cannot make changes to the setup.
 - System configurator: User role can view system configuration and make changes to the setup in the Configuration portal.
 - System configuration - View only: User role can view the system configuration in the Configuration portal, but cannot make changes to the setup.

Once the System Admin Indicator roles have been configured, the configurations and roles are available in both the Configuration Panel and the Administration Portal.

Rule Detail - Rule Type: Authorization Profile

- *Rule Type: Authorization Profile*: A collection of objects, or roles, such as Technician or Supervisor.
- *Profile*: Authorization profile in SAP. To search for an authorization profile, click on the white box icon to the right of the Profile field. The Profile Selection window displays.
- *Text*: Brief description of the authorization profile. This is a non-editable field.

Rule Detail - Rule Type: Authorization Object

- *Rule Type: Authorization Object*: Baseline object used across mobile applications
- *Authorization Object*: Authorization object in SAP.
- *Authorization Field*: Standard SAP authorization object name
- *Authorization Field*: Free-text field. Text entered depends on developer implementation in SAP.

2.3.15.2 Security Settings - Product Security

Use this tab to configure security settings for a specific mobile application.

Security Settings - Product Security

Security Rule Settings

System Security | **Product Security** | Syclo Class Handler Security

Security Check Rule List

Add Rule Delete Rule

Product	Rule Type	Object Name	Authorization Field	Authorization Field Value
SAP_WORK_MANAGER_00	User Role			

Rule Detail

Product Info
Product:

Security Rule Type
Rule Type:

Select A User Role
Role:
Name:

Security Check Rule List

- *Add Rule button*: Press *Add Rule* to add a new product security rule. Fill in the fields in the Rule Detail section to automatically fill in the fields in this table.

- *Delete Rule button*: Press *Delete Rule* to delete an product security rule. To delete a product security rule, press the rectangle to the left of the Rule Type column in the row you wish to delete and press *Delete Rule*.
- *Product*: This read-only field displays the mobile application chosen in the Rule Detail section.
- *Rule Type*: This read-only field displays the Rule Type from the Rule Detail section.
- *Object Name*: This read-only field is populated from the Profile field in the Rule Detail section if rule type Authorization Object is chosen.
- *Authorization Field*: This read-only field is populated from the Authorization Field selection if rule type Authorization Object is chosen. Standard SAP authorization object name.
- *Authorization Field Value*: This read-only field is populated from the Field Value selection if rule type Authorization Object is chosen.

Rule Detail - Security Rule Type: User Role

- *Product*: Select the mobile application that will contain the product security rule.
- *Rule Type*: User Role: In addition to the standard SAP user profile rules, the user role can add restrictions on what a user can or cannot see in menus or other mobile application screens.
- *Role*: Select a user role contained within SAP.
- *Name*: After the user role is selected, this non-editable field is filled in with the descriptive name of the user role.

Rule Detail - Security Rule Type: Authorization Profile

- *Product*: Select the mobile application that will contain the product security rule.
- *Rule Type*: Authorization Profile: A collection of objects, or roles, such as Technician or Supervisor.
- *Profile*: Select an authorization profile contained within SAP.
- *Text*: After the authorization profile is selected, this non-editable field is filled in with the descriptive name of the authorization profile.

Rule Detail - Security Rule Type: Authorization Object

- *Product*: Select the mobile application that will contain the product security rule.
- *Rule Type: Authorization Object*: Baseline object used across mobile applications.
- *Authorization Object*: Select an authorization object contained within SAP.
- *Authorization Field*: Select a field contained within SAP.
- *Field Value*: Free-text field for additional object configuration. Text entered depends on developer implementation in SAP.

- *Class Handler*: This read-only field is populated from the Data Object Handler field in the Rule Detail section.
- *Class Method*: This read-only field is populated from the Handler Method field in the Rule Detail section.
- *Rule Type*: This read-only field displays the Rule Type from the Rule Detail section.
- *Object Name*: This read-only field is populated from the Profile field in the Rule Detail section if rule type Authorization Object is chosen.
- *Authorization Field*: This read-only field is populated from the Authorization Field selection if rule type Authorization Object is chosen. Standard SAP authorization object name.
- *Authorization Field Value*: This read-only field is populated from the Field Value selection if rule type Authorization Object is chosen.

Rule Detail - Security Rule Type: User Role

- *Data Object Handler*: Select the desired class handler from the drop-down list.
- *Handler Method*: Select the desired handler method from the drop-down list.
- *Rule Type*: User Role: In addition to the standard SAP user profile rules, the user role can add restrictions on what a user can or cannot see in menus or other mobile application screens.
- *Role*: Select a user role contained within SAP.
- *Name*: After the user role is selected, this non-editable field is filled in with the descriptive name of the user role.

Rule Detail - Security Rule Type: Authorization Profile

- *Data Object Handler*: Select the desired class handler from the drop-down list.
- *Handler Method*: Select the desired handler method from the drop-down list.
- *Rule Type*: Authorization Profile: A collection of objects, or roles, such as Technician or Supervisor.
- *Profile*: Select an authorization profile contained within SAP.
- *Text*: After the authorization profile is selected, this non-editable field is filled in with the descriptive name of the authorization profile.

Rule Detail - Security Rule Type: Authorization Object

- *Data Object Handler*: Select the desired class handler from the drop-down list.
- *Handler Method*: Select the desired handler method from the drop-down list.
- *Rule Type: Authorization Object*: Baseline object used across mobile applications
- *Authorization Object*: Select an authorization object contained within SAP.
- *Authorization Field*: Select a field contained within SAP.
- *Field Value*: Free-text field for additional object configuration. Text entered depends on developer implementation in SAP.

3 Data Distribution Model Overview

A data distribution model defines how and what back end data are downloaded to the mobile devices. It takes into consideration various factors and criteria when determining what back end data should be downloaded to the mobile client and to the mobile user. Some common criteria include:

- Relevant business processes and business rules
- User business roles in the organization
- Last mobile client synchronization time with the back end system
- Back end 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 will be 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.

3.1 Common Filters

The filters listed in this table are common to all SAP Work Manager distribution rules. See the specific rules for details on filter requirements specific to those rules.

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, Mandatory	Any	Restricts work order distribution based on the work order maintenance plant company code.
CO_AREA	Data Distribution, Optional	Any	Restricts work order distribution based on work order maintenance plant controlling area.
DATE_CLOSE	Data Distribution, Optional	Any	Restricts work order distribution based on date range within which work order has been closed.

Filter Name	Type	Value	Comments
DATE_COMPLETION	Data Distribution, Optional	Any	Restricts work order distribution based on date range within which work order has been technically completed.
DATE_RELEASE	Data Distribution, 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	Data Distribution, Optional	Any	Restricts work order distribution based on work order operation level activity type.
OPER_CONTROL_KEY	Data Distribution, Optional	Any	Restricts work order distribution based on work order operation level control key.
OPER_EXCL_SYST_STAT	Data Distribution, Optional	Any	Restricts work order distribution by excluding work orders with operations that have a specified system status code.
OPER_EXCL_USER_STAT	Data Distribution, Optional	Any	Restricts work order distribution by excluding work orders with operations that have a specified user status code.
OPER_INCL_SYST_STAT	Data Distribution, Optional	Any	Restricts work order distribution by requiring work order operations with a specified system status code.
OPER_INCL_USER_STAT	Data Distribution, Optional	Any	Restricts work order distribution by requiring work order operations with a specified user status code.
OPER_PLANT	Data Distribution, Optional	Any	Restricts work order distribution based on the work order operation level plant.
PLANGROUP	Data Distribution, Optional	Any	Restricts work order distribution based on the work order header planner group.
PLANPLANT	Data Distribution, Optional	Any	Restricts work order distribution based on the work order header planning plant.

Filter Name	Type	Value	Comments
PLANT	Data Distribution, Optional	Any	Restricts work order distribution based on the work order header maintenance plant.
PM_PHASE	Data Distribution, Optional	Any	Restricts work order distribution based on the work order maintenance process phase.
WO_EXCL_SYST_STAT	Data Distribution, Optional	Any	Restricts work order distribution by excluding work orders header with a specified system status code.
WO_INCL_SYST_STAT	Data Distribution, Optional	Any	Restricts work order distribution by requiring a work order header with a specified system status code.
WO_EXCL_USER_STAT	Data Distribution, Optional	Any	Restricts work order distribution by excluding a work order header with a specified user status code.
WO_INCL_USER_STAT	Data Distribution, Optional	Any	Restricts work order distribution by requiring a work order header with a specified user status code.
WORK_CNTR	Data Distribution, Optional	Any	Restricts work order distribution based on the work order header work center.
DOC_BDS_ACTIVE	Data Segment, Optional	Data Segment, Optional	Determines whether the BDS attachment is supported.
DOC_DMS_ACTIVE	Data Segment, Optional	Data Segment, Optional	Determines whether the DMS attachment is supported.
DOC_GOS_ACTIVE	Data Segment, Optional	Data Segment, Optional	Determines whether the GOS attachment is supported.
DOC_GOS_RELTYPE	Data Segment, Optional	Data Segment, Optional	Determines whether the GOS attachment is supported based on a GOS relationship.
DOC_TYPE	Data Segment, Optional	Data Segment, Optional	Determines whether the DMS attachment is supported based on the DMS document type.
DOC_LINK_OBJ	Data Segment, Optional	Data Segment, Optional	Determines whether the DMS attachment is supported based on the linked SAP object.
HISTORY_DURATION	Data Segment, Optional	Any	Restricts the number of work order history records returned, based on number of days in the past.
HISTORY_REFERENCE_DT	Data Segment, Optional	Any	Restricts the number of work order history records returned based on a specific date.
INSP_ORDER_TYPE	Data Segment, Optional	Any	Identifies work orders as inspection round-enabled work orders, based on the order type specified.

Filter Name	Type	Value	Comments
NOTIF_EXCL_SYST_STAT	Data Segment, Optional	Any	Restricts notification related to work orders based on the notification header system status.
NOTIF_INCL_SYST_STAT	Data Segment, Optional	Any	Restricts notification related to work orders based on the notification header system status.
NOTIF_EXCL_USER_STAT	Data Segment, Optional	Any	Restricts notification related to work orders based on the notification header user status.
NOTIF_INCL_USER_STAT	Data Segment, Optional	Any	Restricts notification related to work orders based on the notification header user status.
NOTIF_NO	Data Segment, Optional	Any	Restricts notification related to work orders based on the notification number.
NOTIF_TYPE	Data Segment, Optional	Any	Restricts notification related to work orders based on the notification type.
NUM_WO_HISTORY	Data Segment, Optional	Any	Restricts the maximum number of work order history records that can be returned.
NUM_WO_PENDING	Data Segment, Optional	Any	Restricts the maximum number of pending work order records that can be returned.
OBJECT_HIER_LEVEL	Data Segment, Optional	Any	Restricts the level of technical object hierarchy information associated with the work order that is returned.
PENDING_DURATION	Data Segment, Optional	Any	Restricts the number of pending work order records returned, based on the number of pending work orders in the future.

3.2 Work Order Distribution Rules

The standard SAP Work Manager application work order distribution is controlled by the MDO (mobile data object) **SWM65_WORKORDER_GENERIC** *GET* method. 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.

3.2.1 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 (i.e., the technician) must have 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.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method [GET_BEGIN](#): Influence data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE1](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.2.2 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 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	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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method [GET_BEGIN](#): Influence data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE2](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.2.3 Distribution by Work Order Sub-Operation Person Responsible

Requirements

The following are requirements before configuring the distribution model for Distribution by Work Order Sub-Operation 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 sub-operation 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	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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method [GET_BEGIN](#): Influence data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE3](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.2.4 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 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	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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method [GET_BEGIN](#): Influence data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE4](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.2.5 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 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	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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method [GET_BEGIN](#): Influence data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE5](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.2.6 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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method [GET_BEGIN](#): Influence input data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE6](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.2.7 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:

- Pre-defined partner function is used in work order for work assignment
- Employee number is not required
- By default, mobile user's SAP user ID is used for data distribution by partner function
- 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	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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method [GET_BEGIN](#): Influence input data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE7](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.2.8 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 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	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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method [GET_BEGIN](#): Influence input data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE8](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.2.9 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 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	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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_TYPEA*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.2.10 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 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	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, it should be value 30.

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_TYPE9*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.2.11 Distribution by Custom Logic

Custom implementation of the distribution logic is achieved by implementing BAdI / SMERP/MDO_PM_WO_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_OTHERS*: Implement custom logic to determine the work order list that is distributed to the mobile client.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.3 Notification Distribution Rules

There are five assignment types that allow for distribution by notification assignment types in SAP Work Manager.

i Note

Distribution rules for notifications are supported. However, you must also perform customizations in Java and the Agency Editor to implement notifications.

Notification requests are assigned to the technician directly using the *Person Responsible* field or using the work center, planner group, or related business partner of the technician. The SAP Work 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. The out-of-box default uses partner function *VU-User Responsible* as the default partner function and technician *SAP User ID* as the partner number if there is no MAM configuration set up for the user.
- **5 - Header Level Work Center:** Assign this notification to the work center associated with the technician through the header level *Work Center* field.

Customers can choose the appropriate distribution model based on their specific business processes and requirements.

3.3.1 Distribution by Notification Header Person Responsible

Requirements

The following are requirements before configuring the distribution model for Distribution by Notification 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 Notification header partner overview as the Person Responsible
- Notification has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	1	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_NOTIF_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_TYPE1*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.3.2 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:

- Mobile user (i.e., the technician) must have an employee number (personnel number) assigned in SAP
- Employee number is assigned to the Notification task Person Responsible field
- Notification has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	2	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_NOTIF_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_TYPE2*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.3.3 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:

- Mobile user (i.e., the technician) has been assigned to the planner group based on the business
- Employee number is not required
- Planner group is associated with the mobile user is assigned to the Notification header
- Notification has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	3	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_NOTIF_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_TYPE3*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.3.4 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
- Out-of-box, the mobile user's SAP User ID is used for data distribution by partner function using User Responsible
- Employee number is not required
- Notification has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	4	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_NOTIF_BADI.

- Method [GET_BEGIN](#): Influence input data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE4](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.3.5 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:

- 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 the Notification header

- Notification has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	5	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_NOTIF_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_TYPE5*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.3.6 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 On-Demand Notification look-up scenario
- Employee number is not required
- Notification has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	9	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_NOTIF_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_TYPE9*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.3.7 Distribution by Custom Logic (Notifications)

Requirements

The following are requirements before configuring the distribution model for Distribution by Custom Logic:

- Customer can define own distribution logic using BAdI option
- Employee number is not required
- Notification has not been marked for deletion

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	Z	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMERP/MDO_PM_NOTIF_BADI.

- Method *GET_BEGIN*: Influence input data before standard distribution rule execution.
- Method *GET_ASSIGNMENT_OTHERS*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

3.4 Periodic Meter Reading Distribution Rules

There are five Assignment Types that allow for periodic meter readings in SAP Work Manager.

The valid Assignment Types are:

- **1 - Free Search** - Allows wide open searches, using BAPI parameters and ConfigPanel data object filters, if set.
- **2 - Meter Reading Unit** - Allows for distribution by meter reading unit (Transaction E41G, table TE115 meter readers). Class Handler /SMISU/CL_BT_X_MRUNIT_DOF for Filter Rule
SMM21_MRPERIODIC_DOCUMENT.GET.MR_UNIT can be used to determine the meter reading unit for the user.
- **3 - Personnel Number** - Distribution using the user's assigned Personnel Number to determine the meter reading unit (table TE115 meter readers).
- **4 - Business Partner** - Distribution using the Business Partner assigned to the user to get the meter reading unit.
- **5 - BAdI Implementation** - This is a custom implementation where the customer develops code to determine the meter reading unit.

Customers can choose the appropriate distribution model based on their specific business processes and requirements.

3.4.1 Distribution by Free Search (Periodic Meter Readings)

Requirements

The following are requirements before configuring the distribution model for Distribution by Free Search:

- Input parameters need to be passed to BAPI to restrict values

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	1	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMISU/MDO_ISU_MRPERIODICORDER.

- Method `GET_ASSIGNMENT_TYPE`: Influence data before standard distribution rule execution.
- Method `GET_ASSIGNMENT_TYPE1`: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method `GET_END`: Enhance/enrich output work order data returned to the mobile client.

3.4.2 Distribution by Meter Reading Unit

Requirements

The following are requirements before configuring the distribution model for Distribution by Meter Reading Unit:

- Class Handler `/SMISU/CL_BT_X_MRUNIT_DOF` for Filter Rule
`SMM21_MRPERIODIC_DOCUMENT.GET.MR_UNIT` can be used to determine the meter reading unit for the user
- Implementation of enhancement option `/SMISU/MDO_ISU_MRPERIODICORDER~GET_MR_UNIT` required

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	2	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI `/SMISU/MDO_ISU_MRPERIODICORDER`.

- Method `GET_MR_UNIT`: Meter Reading Unit Range for Periodic Meter Reader..
- Method `GET_ASSIGNMENT_TYPE`: Influence data before standard distribution rule execution.
- Method `GET_ASSIGNMENT_TYPE2`: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method `GET_END`: Enhance/enrich output work order data returned to the mobile client.

3.4.3 Distribution by Personnel Number

Requirements

The following are requirements before configuring the distribution model for Distribution by Personnel Number:

- Use the Personnel Number to determine the meter reader from Table TE115-Meter Readers

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	3	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMISU/MDO_ISU_MRPERIODICORDER.

- Method [GET_ASSIGNMENT_TYPE](#): Influence data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE3](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.4.4 Distribution by Business Partner

Requirements

The following are requirements before configuring the distribution model for Distribution by Business Partner:

- Use the Employee Business Partner to determine the meter reader from Table TE115-Meter Readers

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	4	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMISU/MDO_ISU_MRPERIODICORDER.

- Method [GET_ASSIGNMENT_TYPE](#): Influence data before standard distribution rule execution.
- Method [GET_ASSIGNMENT_TYPE4](#): Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method [GET_END](#): Enhance/enrich output work order data returned to the mobile client.

3.4.5 Distribution by Enhancement Spot Implementation

Requirements

The following are requirements before configuring the distribution model for Distribution by Enhancement Spot Implementation:

- Implementation of GET_ASSIGNMENT_TYPE5 method of enhancement spot /SMISU/MDO_ISU_MRPERIODICORDER

Back End Data Filter Matrix

Filter Name	Type	Value	Comments
ASSIGNMENT_TYPE	Data Distribution, Mandatory	5	Defines which distribution model is used

Enhancement Options

The customer can enhance or enrich the standard work order distribution result by implementing BAdI / SMISU/MDO_ISU_MRPERIODICORDER.

- Method [GET_ASSIGNMENT_TYPE](#): Influence data before standard distribution rule execution.

- Method *GET_ASSIGNMENT_TYPE5*: Modify/enrich the result of the work order list returned by the standard distribution rule.
- Method *GET_END*: Enhance/enrich output work order data returned to the mobile client.

4 Configuring SAP Work Manager with LAM

In addition to the basic SAP Work Manager configuration, there are a few considerations when configuring SAP Work 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.

4.1 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:

- Order headers
- Operations
- Notification headers
- Items
- Equipment details
- Equipment characteristics
- Functional locations
- Functional location characteristics

You can create or edit the following with linear data:

- New work orders
- New operations
- New notifications
- New items
- Measurement documents

You can turn the LAM feature on or off for each of the application components individually through the [Client Globals](#) tab of the ConfigPanel. Out of the box, the Linear Asset Management feature is turned on for all

components.

Global Group	Global Name	Global Value	Global Scope	Active Flag
APPLICATION_CONFIG	Equipment.LAM	Y	Application	<input checked="" type="checkbox"/>
APPLICATION_CONFIG	FunctionalLocation.LAM	Y	Application	<input checked="" type="checkbox"/>
APPLICATION_CONFIG	MeasurementDocument.LAM	Y	Application	<input checked="" type="checkbox"/>
APPLICATION_CONFIG	Notif.LAM	Y	Application	<input checked="" type="checkbox"/>
APPLICATION_CONFIG	Wo.LAM	Y	Application	<input checked="" type="checkbox"/>

Set the *Active* flag in the *Client Globals* tab. In addition, define several MDOs and objects for the complex table and data table.

i Note

The MDOs as follows are related to EHP5 and above only.

Access and define the following, as applicable to the components you have turned on:

- LAM_CLASSIFICATION_GENERIC
- LAM_EQUIPMENT_GENERIC
- LAM_FUNC_LOCATION_GENERIC
- LAM_OBJECT_DATA_GENERIC
- LAM_WO_LAM_MEASUREMENT_DOCUMENT
- LAM_EQUIPMENT_COMPOSITE
- LAM_FUNC_LOCATION_COMPOSITE

Complex Table

- LAM_EQUIPMENT
- LAM_FUNCTION_LOCATION
- LAM_OFFSET_TYPE
- LAM_REFERENCE_PATTERN

Data Table

- LAM_OBJECT_TYPE

4.2 Adding a LAM Equipment Type Filter

Configure an equipment type filter for a LAM equipment fetch as well as an exchange process.

Execute transaction code `/N/SYCL0/CONFIGPANEL` to start the SAP Framework Configuration tool, or ConfigPanel. Then, take the following steps to add the new filter for the MDO and the exchange object.

Add the LAM Equipment Type Filter for the Exchange Object SWM65_EQUIPMENT

1. Click the [Exchange Object Configuration](#) link from ConfigPanel home page.
2. Select the object **SWM65_EQUIPMENT** under [Plant Maintenance](#).
3. Select the [Change Detection Condition Filter](#) tab and click the pencil icon from the tool bar to switch to [Change](#) mode.
4. Click the filter [EQUICATGRY](#) under the [Defined Filters](#) section.
5. Click the [Create](#) icon and enter [LAM Equipment Type](#) (for example, [L](#)) into the [Low Value](#) field.
6. Save your changes.
7. If applicable, enter the [Transport Request](#) to include the changes you made.

Add the LAM Equipment Type Filter for MDO Object SWM65_LAM_EQUIPMENT

1. Click the [Mobile Data Object Configuration](#) link from the ConfigPanel home page.
2. Under [CT - Complex Table](#), select the object **SWM65_LAM_EQUIPMENT**.
3. Select the [Data Filter](#) tab and go to the [Standard](#) filter under the [GET](#) method.
4. Click the filter [EQUICATGRY](#).
5. Click the [Create](#) icon and enter the [LAM Equipment Type](#) (for example, [L](#)) into the [Low Value](#) field.
6. Save your changes.
7. If applicable, enter the [Transport Request](#) to include the changes you made.

4.3 Adding Linear Data for Characteristics

The add or edit linear data for characteristics functionality 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 MDO: **SWM65_LAMCLASSIF_VALUE**.

Relevant BAPIs:

- [/SMERP/E5_PM_DOLAMCLASSIF_CRT](#): Adds the LAM classification data
- [/SMERP/E5_PM_DOLAMCLASSIF_UPD](#): Edits the LAM classification data

i 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 is 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.

5 Configuring SAP Work Manager with Customer Service

In addition to the basic SAP Work Manager configuration, there are a few considerations when configuring SAP Work Manager with the Customer Service component using the Add or Edit Business Partner feature.

Configuration for Add or Edit Business Partner Feature

To fully implement the Add or Edit Business Partner feature of the Customer Service component, consider the following MDOs.

- MDO *SCS22_PARTNER_FUNC* returns a list of partners functions that can be edited or added. Its filter, PARTNER_FUNCTION, as delivered, is set to include: Sold-To (AG), Person Responsible (VW), Vendors (LF), and User Responsible (VU).
Due to the available lookups behind these partner functions, additional partner functions should be one of the following partner types: Customer (KU), Vendors (LI), Personnel Number (PE), or User (US).
This MDO is accessed through the BAPI */SMERP/SD_CTPARTNERFUNC_GET*.
- MDO *SCS22_PARTNER_FUNC_DISPLAY* returns the list of all partner functions and is used for display purposes.
This MDO is accessed through the BAPI */SMERP/SD_CTPARTNERFUNC_GET*.
- MDO *SCS22_VENDOR* controls the lookup for vendors. Filters are available to influence this list, including PURCH_ORG, COUNTRY, PLANT, COMP_CODE, and VENDOR_NO. By default, all vendors are returned.
This MDO is accessed through the BAPI */SMERP/MM_CTVENDOR_GET*.
- MDO *SCS22_HR_EMPLOYEE_LIST* controls the lookups for employees and users. Several filters are available to restrict this list. The filters CO_AREA and PERS_AREA are set by default. Their initial settings are set by the user parameters CAC and PBR, respectively.
This MDO is accessed through the BAPI */SMERP/HR_CTEMPLOYEE_LIST_GET*.
- MDO *SCS22_CUSTOMER* controls the lookup for customers. Several filters are available to restrict this list. Initially, only the PARTNER_FUNCTION filter is set, which ensures the customer is set up as Sold-To Partner.
This MDO is accessed through the BAPI */SMERP/CS_CTCUSTOMER_GET*.
- MDO *SCS22_PARTNER_DET_PROC* and *SCS22_PARTNER_DET_PROC_NOTIF* return the partner determination schema for Customer Service orders and notifications. This data is restricted by the PARTNER_FUNCTION filter which should be synchronized with the PARTNER_FUNCTION filter of the MDO *SCS22_PARTNER_FUNC*.
These MDOs are accessed through the BAPIs */SMERP/CS_CTPARTDETRPOC_GET* and */SMERP/CS_CTPARTDET_NOTIF_GET*, respectively.

6 Configuring SAP Work Manager with Meter Management

In addition to the basic SAP Work Manager configuration, there are a few considerations when configuring SAP Work Manager with Meter Management.

Unlike the LAM functionality, Meter Management is not a feature that is turned "on" or "off." Meter Management is activated during the installation process when the ISU install files are selected instead of the standard SAP Work Manager install files.

MDOs to Consider for Meter Management

To fully implement the Meter Management component, consider the following MDOs.

- **SMM22_MRPERIODIC_DOCUMENT:** Valid assignment types include:
 - **1 - Free Search:** Allows wide open searches, using BAPI parameters and ConfigPanel data object filters, if set.
 - **2 - Meter Reading Unit:** Allows for distribution by meter reading unit (transaction `Ⓜ41G`, table TE115 meter readers). You can use class handler `/SMISU/CL_BTX_MRUNIT_DOF` for filter rule `SMM22_MRPERIODIC_DOCUMENT.GET_MR_UNIT` to determine the meter reading unit for the user.
 - **3 - Personnel Number:** Distribution using the assigned personnel number of the user to determine the meter reading unit (table TE115 meter readers).
 - **4 - Business Partner:** Distribution using the business partner assigned to the user to get the meter reading unit.
 - **5 - BAdI Implementation:** Custom implementation where the customer develops code to determine the meter reading unit.
- **SMM22_ISU_WORKORDER_GENERIC:** Replaces the standard work order MDO in ISU systems. The assignment types are the same as the standard work order MDO.
- **SMM22_ISU_NOTIFICATION_GENERIC:** Replaces the standard notification MDO in ISU systems. The assignment types are the same as the standard notification order MDO.
- Additional exchange objects to consider:
 - SMM22_DISCONNECTION_DOCUMENT
 - SMM22_DEVICE SMM21_DEVLOC
 - SMM22_DEVICE_CAT
 - SMM22_CONNOBJ
 - SMM22_INSTALL
 - SMM22_METER_READING_DOCUMENT
 - SMM22_PREMISE
 - SMM22_REGISTER

7 Common Changes for Work Manager

The SAP Work Manager application can be modified in several ways from the default configuration provided by the installers. Any modifications can require changes to the application project using the Agency Editor, the Java synchronization logic in the Eclipse Java project, or the configuration panel in the Mobile Add-On for ERP. Changes are made to add or enable additional functionality, to disable or remove functionality, or to alter the behavior of default features and functionality.

During implementation, there will be changes made in almost all environments. The default implementation of SAP Work Manager includes the common functionality required of most implementation environments. During implementation it will likely be necessary to change certain configuration options to match the needs of the specific site. Such common configuration changes are made for numerous reasons, including:

- Filtering the data mobile users receive based on work location, skill set, statuses, or assignment methodology.
- Altering how data is updated to the SAP system based on changes captured in on the client devices.
- Retrieving different or additional values for business objects from the default data sets provided with the SAP Work Manager application.

In many cases, these modifications are made prior to moving into a production environment. These changes can also be made post-production. Post-production modifications should be migrated to the production system only after significant testing and according to the standard migration process at the implementation site.

Procedures are provided here for common configuration changes made to the SAP Work Manager application after its initial implementation. Information is provided before each procedure on the general use case for such a change. Note that minor variation from these procedures is acceptable for the specific use case or settings to match the needed requirements.

7.1 Copying an Object to the Customer Namespace

When you copy either a mobile data object or an exchange object to the customer namespace before making configuration modifications, you ensure that you can easily roll back application changes without affecting the original objects.

Context

The following procedure provides information on making a copy of a mobile data object (MDO) or exchange object within the Mobile Add-On for ERP. For configuration changes where you are modifying an MDO or exchange object, we recommend that you make a copy first and place it in the customer namespace. In any of the procedures provided where you are instructed to copy an MDO or an exchange object, refer to this procedure for instructions.

Once you copy and modify an MDO and that copy is then modified for a configuration change, you may need to adjust the BAPI wrapper assignment to reference the new MDO. 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 Mobile Add-On for ERP.
2. Click either *Exchange Object Configuration* or *Mobile Data Object Configuration* from the navigation menu.

The Data Object Detail panel opens.

i Note

Figures shown in this procedure are taken from the Mobile Data Object Configuration page. Screens can look different when configuring an exchange object.



3. In the list of MDOs or exchange objects, select the one to copy.



4. Click the *Copy* button.
5. In the *Basic Data* section, in the `<Mobile Data Object ID>` field, add a *Z* to the beginning of the object name.



6. When finished, click the *Save* button to save the object copy.

A copy of the original object is created in the customer namespace. You can now modify the object, with the backup object available for rollback purposes, if necessary.

7.2 Working with BAPI Wrappers

BAPI wrappers are an element type within the Mobile Add-On for ERP. The BAPI wrappers are the elements through which calls are made by the SAP Mobile Platform to the elements configured in the MDOs (ex: the `GET`, `CREATE`, `UPDATE`, and `DELETE` methods). A BAPI wrapper is assigned to an MDO in the Mobile Add-On for ERP. The BAPI wrapper defines the inputs and outputs for the calls made to it and the data it returns to the SAP Mobile Platform.

After you modify or make a configuration change to an MDO, or when a copy of the MDO is made in the customer namespace as a part of the modification, change the assignment settings of a BAPI wrapper to reference the new copy.

7.2.1 Changing the MDO Assignment of a BAPI Wrapper

Context

Once you create an MDO or an MDO copy, assign it to a BAPI wrapper. During runtime, the MDO triggered is determined based on BAPI wrapper assignment.

The following procedure describes the steps to change the MDO assignment of a BAPI wrapper.

Procedure

1. Open the Mobile Add-On for ERP ConfigPanel through SAP.
2. Click the *BAPI Wrapper Configuration* menu item.
3. Navigate to and highlight the BAPI wrapper that will use the new or modified MDO or Z-MDO in the BAPI Wrapper List tree.



4. Click the *Assignment* tab.
5. Click the *Change* button.
6. Use the *Add* button in the *Mobile Data Object Assignment List* panel to add the new Z-MDO.
7. Click the arrow to the right of the *<Mobile Data Object ID>* field and select the desired MDO from the dropdown list.



8. Change the *<Method Type>* field to select the MDO method called by the BAPI wrapper.
9. Highlight the original MDO in the *Mobile Data Object Assignment List* table and click the *Delete* button.

The original MDO is no longer assigned to the BAPI wrapper, leaving the newly added Z-MDO.

10. Click the *Save* button to save the changes.

7.3 Working with MDO Filter Rules

Data filters are a part of an MDO's configuration, most commonly related to the `GET` method's behavior. For many of the configuration changes made to SAP Work Manager, it is necessary to adjust the rules for one or more of these filters.

Filter rules specify a single field within the database tables from which data is being retrieved, and the condition(s) under which records should be included in the method's operation based on that field's value. Many of the filters in SAP Work Manager either do not contain any active rules or contain rules that may need to be adjusted. A filter only has an effect on the synchronization behavior when it has one or more active rules.

Here, the general procedure is provided for adjusting a filter rule in the Configuration portal of the Mobile Add-On for ERP. When making configuration changes, reference this procedure to change filter rule settings.

7.3.1 Changing MDO Filter Rules

Context

Many of the common configuration changes made for an implementation involve the modification or addition of one or more filter rules within an MDO. In SAP, 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 another 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.

This procedure describes the steps necessary to modify a data filter rule for an MDO. The specific settings of a given rule vary depending on the overall nature of the change being made. Subsequent procedures reference this process and provide the detailed values and settings for the filter rules involved in the specific change.

Procedure

1. Open the Mobile Add-On for ERP ConfigPanel through SAP.
2. From the ConfigPanel Home page, click the *Mobile Data Object Configuration* menu item.

The Mobile Data Object Configuration page displays.

3. Expand the *Mobile Data Object* tree in the *Data Object Navigation Tree* section and highlight the desired Z-MDO created through the *Copying an Object to the Customer Namespace [page 102]* procedure.
4. Expand the desired method in the *Defined Filters* list.
5. Under the method, there are listed all current filters defined for the method. Select the filter to which you are modifying the rules.

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.

6. To add a new rule, edit an existing one, or delete a rule from the filter, click the *Change* button.

Many of the fields in the *Rule Editor* section become editable, and two new buttons are displayed to the right of the *Rule List* section: *Create* and *Delete*.

7. To delete a rule from the filter, select that rule in the list and click *Delete*. If you are making no further changes, click the *Save* button to completely remove the rule from the filter. If you are making additional changes, save the filter after completing all changes.
8. To add a new rule to the filter, click the *Create* button. To edit an existing rule within the filter, select it in the *Rule List*. The current settings for the rule display in the <DOF Rule Type>, <Sign>, <Option>, <Low Value>, <High Value>, and <Active Flag> fields within the *Rule Editor* section.
9. Set or modify the editable fields according to the needs of the application. For a detailed description of all fields, see the topic.

10. Set the <Active Flag> field is set to *True* for each added or edited field before saving changes. Inactive filter rules have no effect on the synchronization processing.
11. When you have finished all desired changes to the filter rules, click *Save* to apply your changes.

7.4 Work Order Distribution by Order Type

In the default configuration of SAP Work Manager, work orders are distributed to technicians based on basic assignment parameters. By default, the order type is not included in the synchronization logic, as order types are implementation- and environment-specific. However, it is possible to distribute work orders to users based on the order type.

In many environments, one or more order types are added to SAP ERP specifically for work orders that are distributed to mobile technicians. These order types indicate that only specified work orders should be downloaded by SAP Work Manager. To support this distribution method (for example, only retrieving work orders with a mobile-specific order type), change the data filter rules of the MDOs within the Mobile Add-On for ERP involved in work order synchronization. MDOs include:

- SWM65_PM_ORDER_TYPE
- SWM65_PLANNING_PLANT_ORDER_TYPE
- SWM65_WORKORDER_GENERIC

Each of the MDOs contains a filter named ORDER_TYPE that by default contains no rules. A filter without rules does not affect the data being retrieved by the GET method of the MDO. To change the behavior of the filter to support work order distribution by order type, add a rule to the ORDER_TYPE filter for each order type included in the download.

Once you change the data filter rules of the MDOs, only work orders with one of the specified types are downloaded by the SAP Work Manager application. Other work orders are excluded from the download, even if they are otherwise assigned to the technician performing the synchronization.

7.4.1 Configuring Work Order Distribution by Order Type

Prerequisites

Address the following items before to performing this procedure:

- The order types for work orders that are downloaded to technicians using the SAP Work Manager application are already determined, as they are used in this procedure.
- The person performing this procedure must have access to the Mobile Add-On for ERP Configuration Panel and have permissions to change configuration settings.

Context

The following procedure provides instructions for modifying the synchronization behavior of the SAP Work Manager application so that only work orders with a given order type or types are downloaded to the SAP Work Manager clients of your users. The procedure involves changing the ORDER_TYPE filter in the MDOs involved in work order synchronization. Specifically, rules are added to this filter in each MDO to include only the desired order types. One rule is added to the ORDER_TYPE filter for each order type to be included. If you don't create a rule for a work order type, those work order types are excluded from the work order download synchronization processing. If the work orders are excluded from synchronization processing, it means that the work orders are not present on the mobile clients of your users.

Procedure

1. Access the Mobile Add-On for ERP Configuration Panel and view the ConfigPanel Home page. Select the link [Mobile Data Object Configuration](#).
2. Within the MDO configuration page, expand the node *CT - Complex Table* in the tree panel on the right. In the list of complex table MDOs provided, select the item **SWM65_PM_ORDER_TYPE**.

The settings for the **SWM65_PM_ORDER_TYPE** are displayed in the configuration page.

3. In the configuration page for this MDO, select the *Data Filter* tab, then find the *Defined Filters* list and expand the *Get Method* node.
4. View the rule list for this filter, which is empty in the default configuration of SAP Work Manager.
5. Create a rule for each order type included in the work order distribution to the SAP Work Manager mobile technicians. The settings for each rule are as follows:
 - **DOF Rule Type:** Static Value In Range Format
 - **Sign:** Inclusive
 - **Option:** =
 - **Low Value:** The desired order type

For details on adding and editing filter rules see the topic [Changing MDO Filter Rules \[page 105\]](#).

6. Save your changes once you have added all rules to the filter.
7. Select the **SWM65_PLANNING_PLANT_ORDER_TYPE** MDO, which is also under the *CT - Complex Table* tree panel on the left of the page. Then select the *Data Filter* tab for the MDO.
8. In the *Defined Filters* list, expand the *Get Method* node and select the *ORDER_TYPE* filter.
9. Similar to the previous MDO filters, add one rule for each order type to be included in the distribution to the SAP Work Manager clients. Configure each rule with the following settings:
 - **DOF Rule Type:** Static Value In Range Format
 - **Sign:** Inclusive
 - **Option:** =
 - **Low Value:** The desired order type
10. After you add the rules to this filter, save the changes.
11. Collapse the *CT - Complex Table* node and expand the *DO - Standard Data Object* node. In the MDO list, choose **SWM65_WORKORDER_GENERIC**, then select the *Data Filter* tab.

12. In the *Defined Filters* list, expand the *Get Method* node and select the *ORDER_TYPE* filter. View the rule list for this filter, which is empty in the default configuration of SAP Work Manager.
13. As with the previous *ORDER_TYPE* filters, add one rule for each order type to be included in the distribution to the SAP Work Manager clients. Configure each rule with the following settings:
 - **DOF Rule Type:** Static Value In Range Format
 - **Sign:** Inclusive
 - **Option:** =
 - **Low Value:** The desired order type
14. After you add the rules to this filter, save the changes.

Results

With the completion of this procedure, work orders are downloaded by the SAP Work Manager application only if their order type is set to a type for which a filter rule was created. Other work orders are not retrieved by SAP Work Manager.

Next Steps

Test the modifications thoroughly at this point, verifying that the proper work orders are downloaded to the SAP Work Manager clients based on the order type of the work orders. The end result should be that regardless of other assignment or distribution criteria, only work orders with the order types configured are downloaded.

7.5 Filtering Equipment Complex Table Records by Equipment Status

Data stored on the SAP Work Manager client includes equipment records. These records are stored in a complex table definition. The number of records at a given implementation can be large, depending on the amount of equipment in the database. By default, the SAP Work Manager application filters these records through a user-dependent rule based on the planning plant of the user.

In some environments, a large set of equipment records can affect the performance of the SAP Work Manager client. In these environments, employ additional filtering. One common option is to filter records based on the status of equipment.

To filter records on the status of equipment retrieved for the complex table stored on the SAP Work Manager client, modify the **SWM65_EQUIPMENT** MDO. Configure the *EQU_INCL_SYS_STAT* filter with a rule specifying which status or statuses to include. After you configure the filter rule, only the equipment records with the specified status are retrieved by the SAP Work Manager application for download to the clients.

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

For a given 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 **SWM65_EQUIPMENT** MDO.

7.5.1 Configuring an Equipment Status Filter for an Equipment Complex Table

Prerequisites

Address the following items before performing the procedure:

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

Context

Use the following procedure to create a filter rule for the **SWM65_EQUIPMENT** MDO. You are adding a rule to the existing filter, **EQUI_INCL_SYST_STAT**. Adding a filter rule results in only the equipment records with a status matching the one configured in the rule downloaded to the SAP Work Manager client.

Procedure

1. Within the Mobile Add-On for ERP home page of the ConfigPanel, select the [Mobile Data Object Configuration](#) link.

The Mobile Data Object Configuration page displays.

2. Using the [Data Object Navigation Tree](#), expand the **CT - Complex Table** node. Under this node, select the **SWM65_EQUIPMENT** MDO.

The **SWM65_EQUIPMENT** MDO information displays on the configuration page.

3. Copy the MDO to the customer namespace, naming the new MDO **Z_SWM65_EQUIPMENT**. All references to the MDO in this procedure pertain to this new copy in the customer namespace. Do not modify the original MDO.

For information on copying an MDO, see the procedure [Copying an Object to the Customer Namespace \[page 102\]](#).

4. Once you have copied the MDO, view the new copy by selecting it in the [Data Object Navigation Tree](#).
5. Select the [Data Filter](#) tab.
6. In the [Defined Filters](#) list, expand the node for the [Get Method](#). Select the **EQUI_INCL_SYST_STAT** filter.
7. Add a rule to the filter with the following configuration settings:

- **DOF Rule Type:** Static Value In Range Format
 - **Sign:** Inclusive
 - **Option:** =
 - **Low Value:** Equipment status to filter on
 - **Active Flag:** Checked
8. To include additional statuses in the filter, repeat the previous step for each of the statuses.
 9. Once you add all filter rules, save the changes to the MDO. Then, change the assignment of the BAPI wrapper `PM_CTEQUIPMENT_GET` to reference the MDO in the customer namespace containing the modifications you made.

For information and instructions on changing the MDO assignment of a BAPI wrapper, see the procedure [Changing the MDO Assignment of a BAPI Wrapper \[page 104\]](#).

Results

With the completion of this procedure, the equipment records downloaded by the SAP Work 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 in this procedure are performing as expected before creating additional filters for this same data set. Regardless of additional changes, test the synchronization of the equipment data thoroughly once modifications are made.

7.6 Work Order Assignment Type Options

By default, the SAP Work Manager application determines the assignment of a work order using the personnel number of the work order header. However, implementation environments in different industries or business types may use a different model from the default to determine the proper technician assignment for a work order. The SAP Work Manager application supports several work order assignment models, with a minor configuration change made for the specific model used.

The following models are supported with configuration changes:

- **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:** Suboperation level personnel number of the work order
- **Assignment type 4:** Personnel number for the Work center capacity requirement level
- **Assignment type 5:** Header level of the planner group*
- **Assignment type 6:** Work order operation or task level of the work center*
- **Assignment type 7:** Header level of the business partner*

- **Assignment type 8:** Header level of the work center*

* These assignment types do not require a technician to have a personnel number.

To change the assignment type used in a deployment, modify the Mobile Add-On for ERP MDO within the **SWM65_WORKORDERGENERIC** Configuration Panel.

To configure the assignment type, modify a filter rule in each of the MDOs. Then modify the client global *Wo.AssignmentType* so that its numeric value matches the assignment type number.

i Note

When assignment type changes, you can only configure one assignment type option.

Change the assignment type through the ConfigPanel. There are two places in the ConfigPanel where you change the assignment type value.

- Navigate to ► *Mobile Application Configuration* ► *Client Globals* ►, and look for the global name *Wo.AssignmentType*. Enter a value between 1 and 8, as defined by the assignment type model explanations in this topic.
- Navigate to ► *Mobile Data Object Configuration* ► *DO - Standard Data Object* ►. Select **SWM65_WORKORDER_GENERIC** and then select *Data Filter*. Select ► *Get Method* ► *Get* ► and set the filter *WO_ASSIGNMENT_TYPE* to a value between 1 and 8, as defined by the assignment type model explanations in this topic.

Configuring Event-based Push for Work Orders

Push is only supported for work order assignment types 1-3. It is not supported for assignment types 4-8. You can only configure push for one assignment type at a time.

To configure push for assignment type 1 (level of the person responsible for the work order), make the following changes:

- Set the *Assignment Type Global* = 1
- Set the *PUSH.WorkordersPushForEvent* = **SWM65_EMERGENCYWORKORDER_PUSH**
- Set the **SWM65_WORKORDER_GENERIC** of the mobile data object *WOASSIGNMENT_TYPE* data filter = 1
- Check the *Active Flag* for the **SWM65_EMERGENCY_WORKORDER_PUSH** push scenario and uncheck the *Active Flag* for the other two push scenarios
- Check the *Active Flag* for the **SWM65_WORK_ORDER_PUSH** in EFI Assignment for *SYCLO/PM_EFI_AUFNR_EX_INCL*.

To configure push for assignment type 2 (personnel number assigned to the work order operation), make the following changes:

- Set the *Assignment Type* global = 2
- Set the *PUSH.WorkordersPushForEvent* = **SWM65_EMERGENCY_WORKORDER_OPERATION_PUSH**
- Set the **SWM65_WORKORDER_GENERIC** of the mobile data object *WOASSIGNMENT_TYPE* data filter = 2
- Check the *Active Flag* for the **SWM65_EMERGENCY_WORKORDER_OPERATIONPUSH** push scenario and uncheck the *Active Flag* for the other two push scenarios
- Check the *Active Flag* for the **SWM65_EMERGENCY_WORKORDER_OPERATION_PUSH** in the *EFI Assignment* for */SYCLO/PM_EFI_AUFNR_EX_INCL*

- Set the filter *OPER_PARENT_UPDATE = X* for the **SWM65_WORK_ORDEROPERATIONPUSH** in the *Exchange Object Configuration* for plant maintenance.

To configure push for assignment type 3 (person responsible for the work order), make the following changes:

- Set the *Assignment Type* Global = 3
- Set the *PUSH.WorkordersPushForEvent = SWM64_EMERGENCYWORKORDER_SUBOPER_PUSH*
- Set the **SWM65_WORKORDER_GENERIC** of the mobile data object *WOASSIGNMENT_TYPE* data filter = 3
- Check the *Active Flag* for the **SWM65_EMERGENCY_WORKORDERSUBOPER_PUSH** push scenario and uncheck the *Active Flag* for the other two push scenarios
- Check the *Active Flag* for the **SWM65_EMERGENCY_WORKORDER_SUBOPERPUSH** in the *EFI Assignment* for */SYCLO/PM_EFI_AUFNR_EX_INCL*

Session Timeout Configuration for Push Connections

You can set the client global value to *SessionTimeout* to ensure that unused push connections are not left open indefinitely. Set the client global value in the *Client Globals* tab of the Config Panel. It is measured in seconds. The default setting is 20 sec. After a session timeout, connections are re-established as they are needed.

7.7 Work Order STARTED Status Mapping in SAP ERP Overview

By default, the SAP Work Manager application maps the work order *STARTED* status on the client to the *REL* status in SAP ERP. For many implementations, customers may want to use the *MOBI* status in SAP ERP to indicate that the work order was started on the client and cannot be modified in SAP ERP.

You can map the mobile status to a different status in SAP ERP, by changing the system status technical code for the *STARTED* mobile status for SAP Work Manager. When the system status technical code is changed, updates to SAP ERP made when a work order is started set the status in SAP ERP to the *MOBI* status, matching the entered technical code.

To configure the *STARTED* status mapping, you will work in the *Mobile Application Configuration* screen of the ConfigPanel.

7.7.1 Change the Mapping of Mobile Status STARTED

Prerequisites

Address the following items before performing the procedure:

- Determine and note the technical code of the work order system status to which you are mapping the mobile status *STARTED*.
- The system status to which the mobile status of *STARTED* is to be mapped is configured as a work order status within SAP ERP.
- Access to the Mobile Add-On for ERP ConfigPanel and permissions to change the configuration settings of the elements are granted to you.

Context

The following procedure describes the steps required to change to which system status the mobile status `STARTED` is mapped. When the procedure is complete, the end result of processing the `StartWorkOrder` transaction by the SAP Mobile Platform is that the status of the work order in SAP ERP is set to the one to which the mobile status is mapped.

Procedure

1. Using the ConfigPanel, navigate to the *Mobile Application Configuration* page. In the *Defined Mobile Applications* list, select your application.
2. Select the *Mobile Status Setting* tab. Information displayed on this page includes the section *Mobile Status Mapping*. The *Mobile Status List* table is included in this section. Select the record in this list with an *Object Type* of `WORKORDER` and a *Mobile Status* of `STARTED`.
3. Click the *Change* button to change this record.
4. Change the value of the `<System Status>` field to the technical code of the system status to which you want the `STARTED` mobile status to map to. After your change is made, click the *Save* button to apply this changes.



Results

With the completion of this procedure, the mobile work order status of `STARTED` is mapped to a different system status than the default `REL` status. Processing a `StartWorkOrder` transaction at runtime updates the target work order in SAP ERP with the newly mapped status.

Next Steps

Test your modification to verify when a work order is started on the SAP Work Manager client the resulting status in SAP ERP is set to the status configured in this procedure. Once testing is successful, migrate the modifications made to the production system according to migration processes in place at the implementation site.

7.8 Retrieving Additional Data for Mobile Data Objects Overview

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

Commonly, these additional values are either stored as properties within objects, or as fields within the complex table records. For either case, the changes made to retrieve the new values are similar:

1. Select the additional field or fields that are returned by the *GET* method of the MDO within the MDO that pertains to the mobile application definition.
2. Modify the proper Java class to map the new data value between the BAPI wrapper and the Agentry definition.
3. In the Agentry application project, add either a new object property or a complex table field to store each of the new values that are retrieved.
4. Where necessary, modify the Agentry application project to display the new value, as well as allow for any other interactions, such as search or selection.

7.8.1 Adding New Values to Retrieve for Mobile Data Objects

Prerequisites

Address the following items before performing the procedure:

- Determine and note the desired values to add to the values already retrieved. Also note the tables within which the MDOs reside within SAP ERP.
- The person performing this procedure has access to the ConfigPanel and has permissions to change the configuration settings of the elements within it.

Context

Use the following procedure to add new fields to complex tables or objects. The procedure uses a complex table as an example. However, these steps accomplish the same goal to add new values to object definitions as well. Where there are differences, they are noted in this procedure.

Procedure

1. Navigate to the *Mobile Data Object Configuration* page. Select the proper MDO from the *Data Object Navigation Tree*.

The general settings of the selected MDO are displayed in the *General Settings* tab.

2. Copy this MDO to the Z namespace. Make all changes to the MDO to a new copy in the Z namespace. Do not modify an original, out of the box, MDO.

For information on copying an MDO, see the procedure .

3. Select the additional fields in the MDO:
 - a. Navigate to the *Mobile Data Object Configuration* screen and find your newly created MDO.
 - b. Click the *Result Set Field Selection* tab and click the *Change* button.
 - c. In the *Field Selection Detail* pane, expand the table to locate where to add the new active fields.

The *Field Active* column displays in white, with check mark boxes enabled.



- d. Scroll through the fields to determine which ones to enable or disable. Checking the *Field Active* box enables the field, while unchecking the box disables the field.
 - e. Click the *Save* button to save your changes.
4. Navigate to the ConfigPanel Home page and select the *BAPI Wrapper Configuration* link, *Assignment* tab. Here, change the BAPI wrapper assignment of the appropriate BAPI wrapper to use the MDO in the Z namespace modified in the previous step.

For information and instructions on changing the MDO assignment of a BAPI wrapper, see the [Changing the MDO Assignment of a BAPI Wrapper \[page 104\]](#) procedure.

5. Complete Java synchronization through Eclipse:
 - a. Open the appropriate Java project in Eclipse for the mobile application.
 - b. Locate the appropriate BAPI wrapper Java file in the project. Find it by locating the BAPI wrapper name that is associated with the MDO in the ConfigPanel. Copy the BAPI wrapper name and perform a file search within Eclipse to locate the correct Java file.
 - c. Open the POJO declaration within the Java file.
 - d. Locate the public strings and the properties. Add new data members to the class matching the fields selected for retrieval in the MDO. Also define the new setter methods for those selected fields.

```
public class Plant extends ComplexTableObject {

    public String ID;
    public String Name;
    public String ValuationArea;
    public String CompanyCode;

    public Plant() {
        super();
    }

    public void setProperties(JCO.Table tbl) throws Exception {
        setID(tbl.getString("WERKS"));
        setName(tbl.getString("NAME1"));
        setValuationArea(tbl.getString("BWKEY"));
        setCompanyCode(tbl.getString("BUKRS"));
    }
}
```

- e. Save and compile the Java code by running the `build-mm.xml` script.

i Note

Compile and test the Java code changes to a development environment on the SAP Mobile Platform before publishing to a production environment.

6. Add the new fields to the complex table or object definition in Agentry:
 - a. Open the SAP Work Manager project in Agentry.
 - b. Navigate to the complex table or object that will contain the newly added fields.
 - c. For complex tables, select the *Fields* tab and click the + icon to add a new field.
 - d. Add the new field using the wizard, naming it to match the name given to the values in the Java class. When done, click *Finish*.
 - e. If necessary, click the *Indexes* tab to add new indexes.
 - f. For objects, select the *Properties* tab and click the + icon to add a new field.
 - g. Add the new property using the wizard, naming it to match the name given to the values in the Java class. When done, click *Finish*.
 - h. Make any other applicable changes to the application that is using the complex table or object, such as displaying the new value or modifying rules to search on the new values.
 - i. Publish the application to the SAP Mobile Platform in preparation for testing. Restart the SAP Mobile Platform as changes were also made to the Java logic in support of the new fields.

Results

With the completion of this procedure, one or more new values are retrieved as a part of the data for the object or complex table definition. These new values may be displayed, edited, searched on, or used on the client.

Next Steps

When the modifications are complete, thoroughly test the application. Verify that the values added are retrieved as expected. Test any functionality related to the newly added values. Once testing is successful, migrate the modifications to the production system according to migration processes in place at the implementation site.

7.9 Enabling the Technical Objects Work List Feature

There are two work lists you can turn on: the functional locations work list and the equipment work list.

Context

SAP Work Manager and its components, Customer Service, and Meter Management, come with the technical objects work list feature turned off by default.

Procedure

1. From the ConfigPanel home page, click the *Mobile Application Configuration* link, *Client Globals* tab.
2. Select your application from the *Defined Mobile Applications* list.
3. Search for the global name *FunctionalLocation.Worklist* and set its value to *Yes*.

RecNo	Global Group	Global Name	Global...	Scope	Dep. RecNo	Active	No Ch...	Comment
000000926	APPLICATION_CONFIG	FunctionalLocation Worklist	Y	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Global to enable the functional location work list.
000000927	APPLICATION_CONFIG	Notif.Priority	3	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	DTGLOBALS define static data and is used in con...
000000928	APPLICATION_CONFIG	Wo.AssignmentType	A	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	DTGLOBALS define static data and is used in con...
000000929	APPLICATION_CONFIG	Wo.PlanningPlant	1000	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	DTGLOBALS define static data and is used in con...
000000930	APPLICATION_CONFIG	Wo.Priority	3	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	DTGLOBALS define static data and is used in con...
000000931	APPLICATION_CONFIG	Wo.Type	PM01	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	DTGLOBALS define static data and is used in con...
000000932	CATSRECORDADD	Lock.CATSRecord	Y	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000933	CATSRECORDEDIT	Lock.CATSRecord	Y	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000934	INBOUND_TRANSACTION_QUEUE	DisplayErrorMessage	Y	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000935	INBOUND_TRANSACTION_QUEUE	FetchErrorMessage	Y	User	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

4. Click the *Parameters* tab. Search for the parameter name *ctLocation* with the parameter group *CT_BAPI_WRAPPER*.
5. Change the value of the *ctLocation* parameter found in the previous step to */SMERP/PM_CTFLOC_LSET_GET*.

Note

If you have LAM enabled, set the value to */SMERP/PM_CTLAMFLOC_LSET_GET*.



6. Click the *Client Globals* tab. For the phone form factor for iPhone and Android Phone, ensure that the additional client globals are active (the *Active* flag and *No runtime Change* flags are checked):

- Group = MAIN_MENU_ITEMS, Name = 5, Value = FunctionalLocation
- Group = MAIN_MENU_ITEMS, Name = 7, Value = Settings



7. Turn on the equipment work list:

- Remain on the *Client Globals* tab.
- Search for the global name *Equipment.Worklist* and set its value to *Yes*.
- Click the *Parameters* tab and search for the parameter name *ctEquipment* in the *CT_BAPI_WRAPPER* group.
- Change value of the *ctEquipment* parameter to */SMERP/PM_CTEQUIP_LSET_GET*.

i Note

If you have LAM enabled, set the value to */SMERP/PM_CTLAMEQUIP_LSET_GET*.

- Click the *Client Globals* tab. For the phone form factor for iPhone and Android Phone, ensure that the following additional client globals are active (the *Active* flag and *No runtime Change* flags are checked):
 - Group = MAIN_MENU_ITEMS, Name = 6, Value = Equipment
 - Group = MAIN_MENU_ITEMS, Name = 7, Value = Settings
8. **Configure the add feature of the technical object templates:** Configuring the technical object templates allows you to create new equipment or functional locations based on a template. The template is defined from the back end mobile configuration. The feature is supported by maintaining default templates by Equipment and Functional Location category.
- Return to the ConfigPanel home page. Then navigate to the *Mobile Data Object Configuration* screen. Select your application in the *<Mobile Application Filter>* field.
 - Under *CT-Complex Table*, select *<APP>_EQUIPMENT_TEMPLATE*.
 - Click the *Data Filter* tab and maintain the templates data under the *EQUI_TEMPLATE* with the *Equipment* category and the *Equipment ID* fields, as shown in the following sample

Handler Info

Mobile Data Object Id: SWM65_EQUIPMENT_TEMPLATE Description: Equipment Template Data
 Data Object Handler: /SMERP/CL_PM_EQUI_TEMPLAT

Data Object Filter Rule Definition

Defined Filters

- Get Method - GET
- Standard Filter
- Filter - EQUI_TEMPLATE*

Table Rule Editor

Method Name: GET Filter Name: EQUI_TEMPLATE
 Reference Table Name: /SMERP/PM_EQUI_TEMPLATE_S
 Table Rule Category: Static Table Rules

EquipCategory: M
 Equipment: 10002861

Rule List

EquipCategory	Equipment
M	10002861
M	TEY-01

9. **Create a functional location template:** From the ConfigPanel, go to *Mobile Data Object Configuration*.
 - a. Under *CT-Complex Table*, select `<APP>_FUNCTION_LOC_TEMPLATE`.
 - b. Remain on the *Data Filter* tab and maintain the templates data under the *FLOC_TEMPLATE* with the `<Functional Location>` category and the `<Functional Location>` ID fields, as shown in the following sample.

The screenshot displays the configuration interface for a mobile data object. It is divided into several sections:

- Handler Info:**
 - Mobile Data Object Id: SWM65_FUNCTION_LOC_TEMPLATE
 - Description: Functional Location Template Data
 - Data Object Handler: /SMERP/CL_PM_FLOC_TEMPLAT
- Data Object Filter Rule Definition:**
 - Defined Filters:** A tree view showing 'Get Method - GET' expanded to 'Standard Filter', which contains 'Filter - FLOC_TEMPLATE*'. The 'Filter - FLOC_TEMPLATE*' filter is selected.
 - Table Rule Editor:**
 - Method Name: GET
 - Filter Name: FLOC_TEMPLATE
 - Reference Table Name: /SMERP/PM_FLOC_TEMPLATE1
 - Table Rule Category: Static Table Rules
 - FuncLocCat.: M
 - Functional loc.: 00-B02
 - Rule List:** A table with two columns: 'FuncLocCat.' and 'Functional loc.'. It contains two rows:

FuncLocCat.	Functional loc.
M	00-B02
M	04-B01

7.10 Working with Rounds

An inspection round collects similar repetitive activities and inspections on many different technical objects. Rounds are executed and confirmed by the mobile user. Confirming the inspection round is a key element of the overall inspection round process.

Rounds are defined and scheduled in the back end. The SAP ERP system allows the user to customize the variant or the range of inspection rounds are downloaded to the mobile device. Technical confirmation of the inspection rounds once the user confirms the rounds on the mobile device is performed through a new entry transaction.

- All orders of type *INSP* are classified as inspection round orders on the mobile device
- All orders of type *PRT* assigned to the equipment and functional locations at an operation level are classified as inspection round orders on the mobile device
- All measurement points associated with an inspection round order are displayed in an easy to use wizard format
- If required, the user can create measurement documents on the client
- If follow up activities are necessary, the technician can create notifications

The Rounds feature of the SAP Work Manager application allows you to accomplish the following:

- View inspection rounds from the work order

- View resources and tools for the rounds
- Confirm inspection rounds
- Record readings and create measurement documents for all points across the work order
- Create notifications during inspection rounds
- Confirm unplanned operations during inspection rounds

There are no on or off settings for inspection rounds. Determination of rounds are driven by back end flags. Rounds work orders are distinguished by the following criteria:

- If an operation has a PRT point in it, then it is considered an inspection rounds order.
- The MDO work order filter <INSP_ORDER_TYPE> qualifies which order types are inspection rounds types.

Navigate to [Mobile Data Object Configuration](#) > [Data Filter tab](#) > [DO - Standard Data Object](#) > [<APP>_WORKORDER_GENERIC](#) > [Get Method - GET](#) > [Data Segment](#) > [Filter - INSP_ORDER_TYPE](#) to access the filter.

i Note

The *INSP_ORDER_TYPE* filter is empty out of the box.



7.11 Configuring Multi-Resource Scheduling (MRS)

Use Multi-Resource Scheduling (MRS) to download work orders to the mobile device based on MRS assignments from the back end. Once the assignments are scheduled in the MRS system, the changes update the work order capacity requirements with the proper personnel assignment.

i Note

The minimum requirement for MRS integration is MRS 8 sp5 or above. Install and configure MRS before installing SAP Work Manager.

The following sample shows the MRS assignment from the MRS Builder:



The MRS assignment is copied to the work order capacity requirement assignment. Once the capacities are updated with the proper technician, the order qualifies for the mobile client under Assignment Type *A* (similar to Assignment Type 4).



Set the corresponding assignment type to *A - Multi Resource Scheduling (MRS)*, under **Mobile Data Object Configuration** > **Data Filter** > **DO - Standard Data Object** > **SWM<X>_WORKORDER_GENERIC** > **Get Method - GET** > **Data Distribution** > **Filter - WO_ASSIGNMENT_TYPE** ⌵:

Handler Info

Mobile Data Object Id: SWM65_WORKORDER_GENERIC Description: Work Order Generic
 Data Object Handler: /SMERP/CL_PM_WORKORDER_I

Data Object Filter Rule Definition

Defined Filters

- Create Method - CREATE
- Delete Method - DELETE
- Get Method - GET
- Data Distribution
 - Filter - COMP_CODE
 - Filter - CO_AREA
 - Filter - DATE_CLOSE*
 - Filter - DATE_COMPLETION*
 - Filter - DATE_RELEASE
 - Filter - EMPLOYEE_ID
 - Filter - ORDERID
 - Filter - ORDER_CATG*
 - Filter - ORDER_TYPE*
 - Filter - PARTNER_FUNCTION
 - Filter - PARTNER_ID
 - Filter - PLANGROUP
 - Filter - PLANPLANT*
 - Filter - PLANT
 - Filter - PM_PHASE*
 - Filter - WORK_CNTR
 - Filter - WO_ASSIGNMENT_TYPE***
 - Filter - WO_EXCL_SYST_STAT*
 - Filter - WO_EXCL_USER_STAT

Rule Editor

Method Name: GET Filter Name: WO_ASSIGNMENT_TYF
 Reference Table Name: /SYCLO/PMREF_STR Reference Field Name: ASSIGNMENT_TYPE
 Data Filter Rule Key: SWM65_WORKORDER_GENERIC.GET.WO_ASSIGNMENT_TYPE
 DOF Rule Type: Static Value In Range Format

Enter Range Value

Sign: Inclusive Option: =
 Low Value: A - Multi Resource Scheduling (MRS)
 High Value:
 Active Flag:

Rule List

Rule No.	Rule Type	Rule Value	Active Flag
00001	RANGE	IEQ1	<input checked="" type="checkbox"/>
00002	USERPARAM	/SMERP/PM_WO_DISTMOD	<input type="checkbox"/>

Create Delete

Then, navigate to the *Exchange Object Configuration* page to configure the following exchange objects:

- ▶ **Plant Maintenance** > **SWM<X>_CAPACITY_REQUIREMENTS**: ⌵ Triggers when the MRS builder pushes assignments to the work order
- ▶ **Plant Maintenance** > **SWM<X>_WORK_ORDER_OPERATION_PUSH** ⌵
- ▶ **Plant Maintenance** > **SWM<X>_WORK_ORDER_PUSH**: ⌵ Works in the same way as **SWM<X>_WORK_ORDER_OPERATION_PUSH**, but is only for the operation level, not the suboperation level

7.12 Disabling and Re-Enabling Attachments

SAP Work Manager supports viewing of master data or transaction data attachments on the mobile device. *Attachments* include documents such as Microsoft Office files, PDFs, and other commonly used business files, including videos, pictures, and audio files. When you select *Attachments*, the details screen displays the attachments that are available for download.

By default, the option to download attachments is turned on, but you can configure the download attachment option through the Configuration Panel based on work order type. If there are any attachments, the initial sync

fetches only limited information regarding the attachments. Upon user request, the attachment is brought down to the device. All attachments are maintained in a central repository and controlled through the ConfigPanel.

There are two options for how you can download attachments. Once the option is set in the ConfigPanel, it is the same for all users. The default setting is synchronous download. After tapping on the attachment, a transmit immediately begins to fetch the document. Once the transmit completes, you are brought to the main document screen to view the document.

The other option is asynchronous download. After tapping on the attachment, a push request is sent to the back end to fetch the document while you are returned to the *Attachments* screen. The document is downloaded in the background. Once the download completes, a notification is sent to the client and a message appears on your screen telling you that the document is ready to view.

To enable asynchronous downloads of attachments, set *Attachment.Synchronous* to *N* in the ConfigPanel and set up push. See the [Configuring Push \[page 124\]](#) procedure and following procedures in this guide for more information.

You can download attachments for any work order. On Android devices, you can only view attachments from the Download folder in the device. You cannot open the attachment from within the application as you can with an iOS device.

Downloading and uploading attachments are supported for:

- Orders (header level)
- Orders (header level asset)
- Order (object level)
- Operations via PRT
- Notifications (header level)
- Notifications (header level asset)
- Item (object level)
- Item via Associated Asset
- Equipment (only DMS is supported)
- Functional Locations (only DMS is supported)

The following document services are supported from the back-end system:

- BDS (Business Document Service) / GOS (Generic Object Services): BDS is only supported for header level work orders and notifications
- DMS (Document Management System)

You can allow or disallow attachments individually by object type from the [Mobile Application Configuration](#) **Client Globals** screen of the ConfigPanel. By default, the attachment feature is ON. On the client side, you can only use one service at a time. Therefore, for header level work orders and notifications, you can select BDS or DMS but you cannot select both.

General Mobile Status Setting Conversion Exit Setting System Components Parameters **Client Globals** User Attributes

Mobile Application Info

Mobile Application: SAP_WORK_MANAGER_65 Release: 6.5
 Mobile App. Desc.: SAP Work Manager 6.5

Client Globals

Client Global List

Import/Export Search:

RecNo	Global Group	Global Name	Global...	Scope	Dep. RecNo	Active	No Ch...	Comment
000000741	APPLICATION_CONFIG	Attachment.MaxUploadSize	500000000	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000742	APPLICATION_CONFIG	Attachment.Synchronous	Y	Application	000000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
000000743	APPLICATION_CONFIG	AutomaticallyNOCOifTECO	N	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This will determine to automatically NOCO all notif...
000000744	APPLICATION_CONFIG	BillOfMaterials.Enable	Y	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000745	APPLICATION_CONFIG	BillOfMaterials.FetchAll.Enable	Y	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000746	APPLICATION_CONFIG	Component.S4HANA		Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000747	APPLICATION_CONFIG	CreateActivityReport	N	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Create Notification (M3 - Activity Report) when co...
000000748	APPLICATION_CONFIG	DMS.DocumentPart	000	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000749	APPLICATION_CONFIG	DMS.DocumentType	DRW	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000750	APPLICATION_CONFIG	DMS.DocumentVersion	00	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

The following sample shows the DMS default parameters for document uploads:

Create Copy Delete Change

General Mobile Status Setting Conversion Exit Setting System Components Parameters **Client Globals** User Attributes

Mobile Application Info

Mobile Application: SAP_WORK_MANAGER_65 Release: 6.5
 Mobile App. Desc.: SAP Work Manager 6.5

Client Globals

Client Global List

Import/Export Search: Document

RecNo	Global Group	Global Name	Global...	Scope	Dep. RecNo	Active	No Ch...	Comment
000000748	APPLICATION_CONFIG	DMS.DocumentPart	000	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000749	APPLICATION_CONFIG	DMS.DocumentType	DRW	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000750	APPLICATION_CONFIG	DMS.DocumentVersion	00	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000751	APPLICATION_CONFIG	DMS.OriginalType	1	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000752	APPLICATION_CONFIG	DMS.StatusExtern	AA	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000753	APPLICATION_CONFIG	DMS.StatusIntern	AA	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
000000754	APPLICATION_CONFIG	DMS.StorageCategory	SAP-SYS...	Application	000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

You can configure different document types (DMS/BDS) for work orders and notification by navigating to the [Mobile Data Object Configuration](#) [Data Filter](#) screen. Here you can configure the MDO:

The screenshot displays the 'Data Object Detail (Display Mode)' for 'SWM65_DMS_BDS_SWITCHES'. The 'Handler Info' section shows the Mobile Data Object ID and the Data Object Handler. The 'Data Object Filter Rule Definition' section includes 'Defined Filters' and 'Table Rule Editor'. The 'Table Rule Editor' shows a 'Filter - DMS_BDS_SWITCHES' rule with the following configuration:

- Method Name: GET_BDS_DMS_CONFIG
- Reference Table Name: /ISMERPICORE_KWDOC_SWITCHES
- Table Rule Category: Static Table Rules
- KW Obj Category: NOTIFICATION
- KW Object Type: M1
- DMS Act Flag: Active
- BDS Act Flag: Inactive

The 'Rule List' table below shows the configuration for various object types:

KW Obj Category	KW Object Type	DMS Act Flag	BDS Act Flag
NOTIFICATION	M1	X	
NOTIFICATION	M2	X	
NOTIFICATION	M3	X	
NOTIFICATION	M4	X	
NOTIFICATION	M5	X	
NOTIFICATION	N1	X	
NOTIFICATION	N2	X	
WORKORDER	PM01		X
WORKORDER	PM02		X
WORKORDER	PM03		X

For work order fetch, set the following MDOs, as applicable to your document service choice:

- WORKORDER_GENERIC
- DOC_BDA_ACTIVE
- DOC_DMS_ACTIVE
- DOC_GOS_ACTIVE
- DOC_GOS_RELTYPE: Use if GOS is active to allow different types of attachments. By default, only attachments and notes are supported
- DOC_LINK_OBJ: Used for DMS
- DOC_TYPE: Used for DMS

7.13 Working with Pushes

7.13.1 Configuring Push

Prerequisites

Use the following procedure to enable push on the SAP Mobile Platform. To check if push is enabled, see the `javaBE.ini` file, `[push_logon]` section. Ensure that `ENABLED=true` and that a user password is set.

Context

The default configuration of SAP Mobile Platform uses qRFC for the default push scenario. The qRFC scenario means that there is no need to schedule a job that runs every few minutes to make push operational. When the back end pushes data to the clients, it sends an XML file using the following structure to the SAP Mobile Platform. See the following example:

```
<SWM65_EMERGENCY_WORKORDER_PUSH>
  <SAPPushNotif
    <app>SAP_WORK_MANAGER_65</app>
    <users>
      <usr SCENARIO_ID="SWM65_EMERGENCY_WORKORDER_PUSH"
        USER_GUID="[GUID ID]" USER_ID="[user ID]"/>
    </users>
  </SAPPushNotif>
</VERIFICATION_CODE>[verification code]</VERIFICATION_CODE>
</SWM65_EMERGENCY_WORKORDER_PUSH>
```

Use the following procedure to create and configure a general push to send out to client devices using your application.

Procedure

1. Using the ConfigPanel, navigate to the *Outbound Trigger Detail* screen, and select the HTTP push trigger under the application for which you are enabling the push.

The screenshot displays the 'Outbound Triggers By Mobile App' configuration screen. The left sidebar shows a tree view of triggers under the application 'SAP_WORK_MANAGER_65'. The main area is titled 'Outbound Trigger Detail (Display Mode)' and contains the following sections:

- General Data:** Includes buttons for 'Create', 'Copy', 'Delete', and 'Change'.
- Basic Data:**
 - Outb. Trigger Id: SWM65_WORKORDER_TRIGGER
 - Outb. Trigger Desc.: HTTP Outbound Trigger for Work Order Push
 - Mobile Application: SAP Work Manager 6.5
- Trigger Handler Info:**
 - Outb. Trigger Handler: /SYCLO/CL_CORE_OTRIG_STD_PUSH : HTTP outbound trigger - standard push
 - Outb. Trigger Type: HTTP based trigger
 - Target Host Name: (empty field)
 - Target Host IP: (empty field)
 - Target Host Port No.: 08380
 - URL Identifier Type: IP Address
 - Web Protocol: HTTP
 - Min. Conn. Time(Sec): 7
 - Check Response:
 - Parameter: (empty field)
- Retry Setting:**
 - Allow Retry:
 - Maximum No. of Retry: 999
 - Retry Wait Period (Seconds): 30
- Activation:**
 - Active Flag:
- Administrative Info:**
 - Created By: (empty field)
 - Creation Time Stamp: (empty field)
 - Last Changed By: (empty field)
 - Changed Time Stamp: (empty field)

2. Click *Change* and make the following modifications:
 - a. Leave the *Target Host Name* empty, as the host is determined automatically.
 - b. If you have multiple SAP Mobile Platforms running on the same host, you may need to change the *Target Host Port Number* from the default of *8282*. The port must match the port configured in the *Agentry.ini* file, [httpxml-2] section, *listenOn* parameter.
 - c. Ensure that the *Active Flag* is set to *True*. *Save* your changes.
3. Still within the ConfigPanel, navigate to *Push Scenario Definition* screen, select your application, then the object to push.

Push Scenarios By Mobile App

- ▼ SAP_WORK_MANAGER_65
 - SWM65_BDS_DOCUMENT_PUSH
 - SWM65_DMS_DOCUMENT_PUSH
 - SWM65_EMERGENCY_WORKORDER_OPERATION_PUSH
 - SWM65_EMERGENCY_WORKORDER_PUSH
 - SWM65_EMERGENCY_WORKORDER_SUBOPER_PUSH
 - SWM65_MRS_WORKORDER_PUSH
 - SWM65_NOTIFICATION_PUSH

Push Scenario Detail (Display Mode)

Create Copy Delete Change

General Data Event Setting Outbound Trigger Subscription Settings

Basic Data

Scenario Id: SWM65_EMERGENCY_WORKORDER_PU Alias:

Mobile Application: SAP Work Manager 6.5

Source Setting

Source Type: Exchange Object

Source Object: SWM65_WORK_ORDER_PUSH

Source Handler: /SYCLO/CL_PM_AUFNR_EX_HNDL

Distribution Setting

Distribution Type: Mobile data object

Distribution Object: SWM65_WORKORDER_PUSH

Distribution Handler: /SMERP/CL_PM_PUSHWORKORDER1_DC

Subscriber Setting

Subscriber Type: All mobile users Validity (Hr): 8

Priority (0 - Highest, 9 - Lowest): 0

Disable Owner Originated Push: Check Mobile Transaction History: History Interval (Seconds): 0

Notification Setting

Email Notification: No Data Package:

Email Subject: Emergency Order Assignment

Email Message (140 chars): An emergency order &OBJKEY& has been assigned to you. Please update your mobile device.

Activation

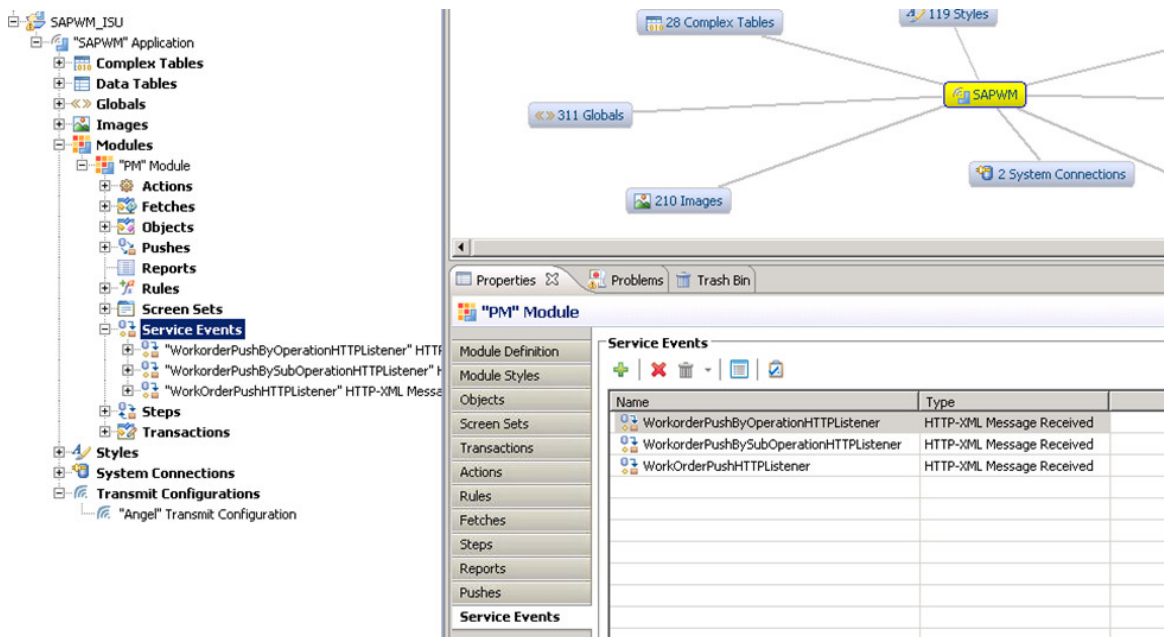
Active Flag: Enable Push History: Require Metadata: Enable Fetch Callback:

Administrative Info

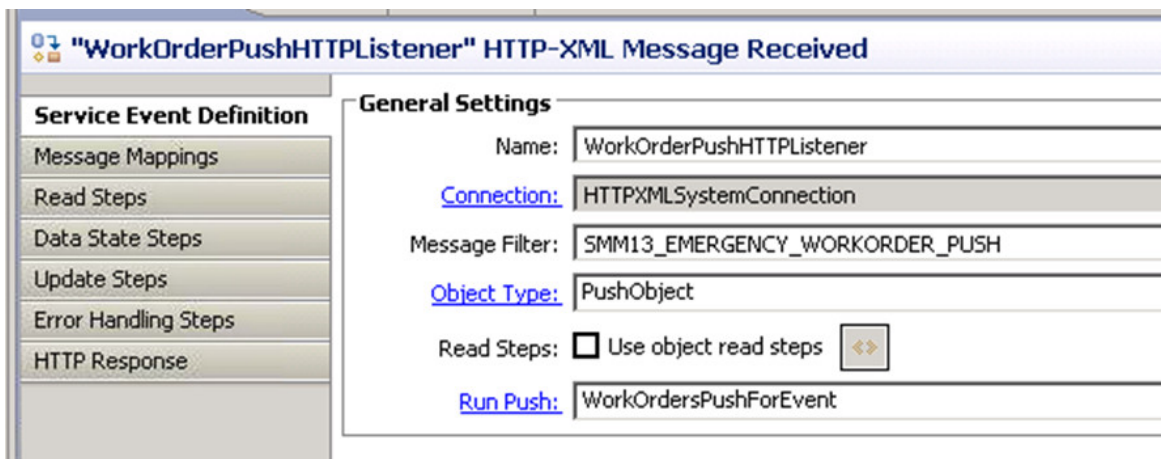
Created By: Creation Time Stamp:

Last Changed By: Changed Time Stamp:

4. All out of the box default settings should be correct. However, ensure that the *Scenario ID* given in the *General Data* tab matches the *Scenario ID* in your Agentry Editor application.
5. Within the Agentry Editor application, navigate to *Service Events* and select the object to push. The object is the same one you selected earlier in the ConfigPanel.

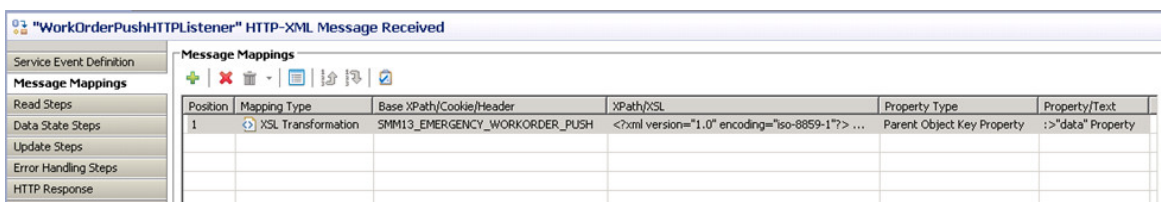


- Double-click the entry to display the *General Settings* and ensure that the entry in the *Message Filter* field matches the *Scenario ID* in the ConfigPanel.

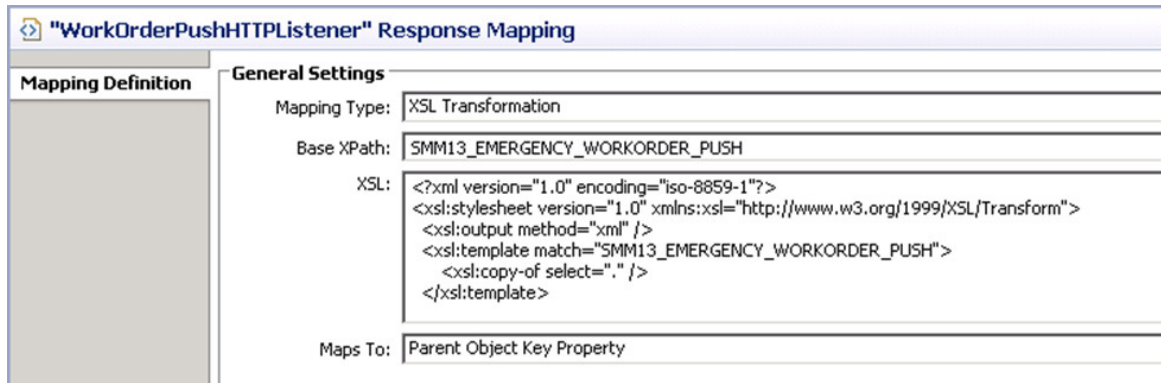


Using this information, the back end sends an XML message to all servers. Based on the *Scenario ID*, the SAP Mobile Platform decides whether the message is relevant. If the message filter does not match the scenario ID, the message is not considered relevant.

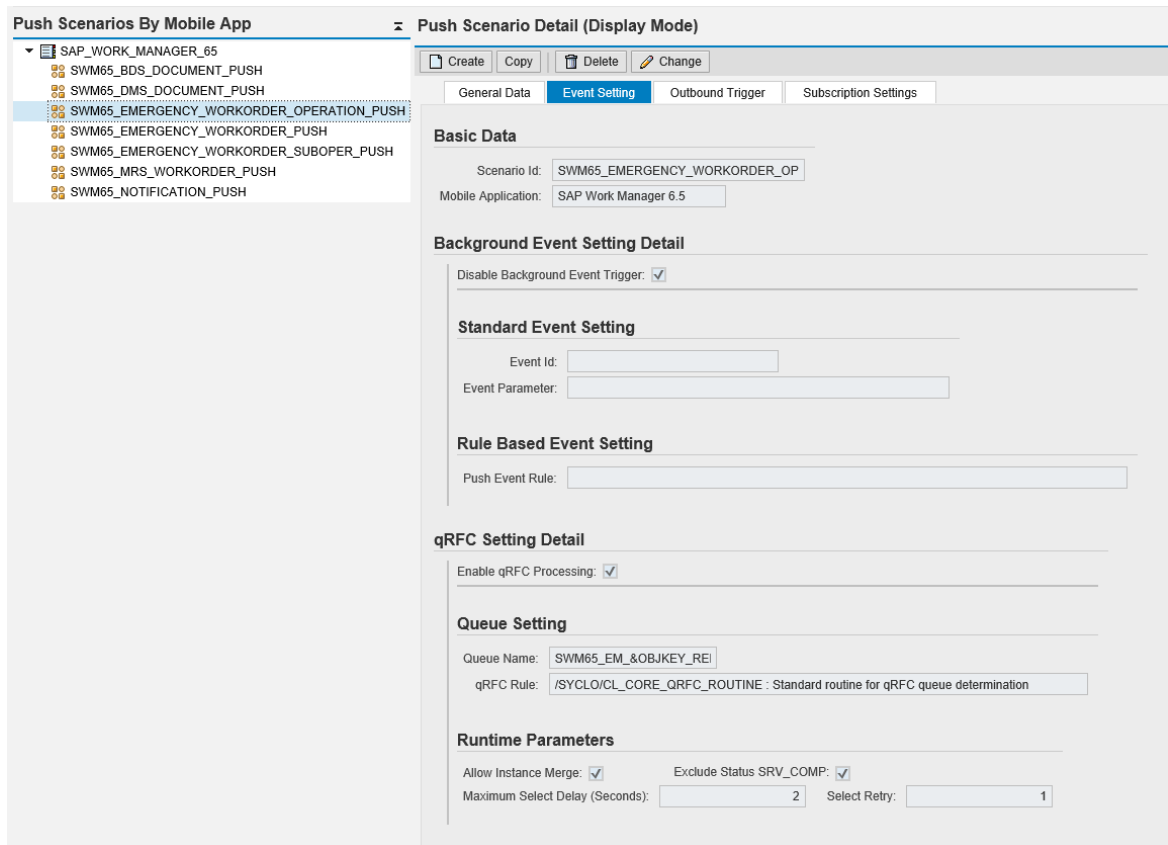
- Click the *Message Mappings* tab.



- Double-click the appropriate entry and ensure that the *Base XPath* and *XSL* use the correct scenario ID from the ConfigPanel.



9. Return to the ConfigPanel and navigate to the *Push Scenario Definition* screen, *Event Setting* tab. Perform the following steps:
 - a. Ensure that *Disable Background Event Trigger* is checked.
 - b. Ensure that *Enable qRFC Processing* is checked. Fill out the fields as applicable to your application.



10. Navigate to the *Outbound Trigger Configuration* screen. Select your trigger associated with the push. Ensure that the *Active Flag* is checked for the trigger.

Outbound Triggers By Mobile App **Outbound Trigger Detail (Display Mode)**

General Data

Basic Data

Outb. Trigger Id: SWM65_WORKORDER_TRIGGER
 Outb. Trigger Desc.: HTTP Outbound Trigger for Work Order Push
 Mobile Application: SAP Work Manager 6.5

Trigger Handler Info

Outb. Trigger Handler: /SYCLO/CL_CORE_OTRIG_STD_PUSH : HTTP outbound trigger - standard push
 Outb. Trigger Type: HTTP based trigger
 Target Host Name:
 Target Host IP:
 Target Host Port No.: 08380
 URL Identifier Type: IP Address
 Web Protocol: HTTP
 Min. Conn. Time(Sec): 7
 Check Response:
 Parameter:

Retry Setting

Allow Retry:
 Maximum No. of Retry: 999 Retry Wait Period (Seconds): 30

Activation

Active Flag:

Administrative Info

Created By: Creation Time Stamp:
 Last Changed By: Changed Time Stamp:

11. Navigate to the *Exchange Object Configuration* screen, *Change Detection Field Selection* tab. Select the appropriate application and push, and ensure that changes on the work or service order are detected on a change.

Exchange Object Info

Exchange Object: SWM65_WORK_ORDER_PUSH Exch. Object Desc.: Work Order Push Exchange
 ExchObject Handler: /SYCLO/CL_PM_AUFNR_EX_HNC

Exchange Object Field Selector

Search:

Field Catalog	Active Flag	Short Description
Table - /SYCLO/MBL01	<input type="checkbox"/>	Mobile device transaction object status
Table - AFAB	<input type="checkbox"/>	Network - Relationships
Table - AFFH	<input type="checkbox"/>	PRT assignment data for the work order
Table - AFFL	<input type="checkbox"/>	Work order sequence
Table - AFFT	<input type="checkbox"/>	Order - Process Instructions
Table - AFFV	<input type="checkbox"/>	Order - Process Instruction Values
Table - AFPO	<input type="checkbox"/>	Order item
Table - AFVC*	<input type="checkbox"/>	Operation within an order
Table - AFVU	<input type="checkbox"/>	DB structure of the user fields of the operation
Table - AFVV	<input type="checkbox"/>	DB structure of the quantities/dates/values in the operation
Table - CAUFVDB*	<input type="checkbox"/>	Document table structure of the order header
Table - IHPA*	<input type="checkbox"/>	Plant Maintenance: Partners
Table - JEST*	<input checked="" type="checkbox"/>	Individual Object Status
Table - NPTX	<input type="checkbox"/>	PS Texts (Network)
Table - OBJK	<input type="checkbox"/>	Plant Maintenance Object List
Table - RESB	<input type="checkbox"/>	Reservation/dependent requirements

Selection Proposal

By Field Name
 By Field Description
 By DDIC Sequence

Results

You have now configured a basic push in the ConfigPanel and in the Agency Editor. Test the push configuration using a small subset of users to ensure that it works.

For a user to receive a push message, the user must be online and not working in offline mode. The application must run actively on the device; running in the background is not sufficient for push to be enabled on the device itself. To see which users are online, use the Agency Editor command line tool and issue the command `users`. Only users whose IDs are shown can receive pushes at that current time.

Next Steps

If you are using SAP Cloud Platform, Mobile Service, also known as SCPms, continue to [Configuring Push with the SAP Cloud Platform, Mobile Service \[page 130\]](#) for additional push configuration steps.

7.13.2 Configuring Push with the SAP Cloud Platform, Mobile Service

Prerequisites

Follow the [Configuring Push \[page 124\]](#) procedure before performing these additional steps to configure a push for the SAP Cloud Platform, Mobile Service.

Procedure

1. From the ConfigPanel home page, click the [Mobile Application Configuration](#) link, then click your mobile application from the [Defined Mobile Applications](#) list. Click the [Client Globals](#) tab.
2. In the globals `<Search>` field, search for `PUSH_NOTIF_CUSTOM_HEADER`. Click the [Change](#) button. Check the [Active Flag](#) box to activate the global.
3. [Save](#) your changes.
4. From the ConfigPanel home page, click the [Push Scenario Definition](#) link, [Outbound Trigger](#) tab. Choose your application from the [Mobile Application Filter](#). Select the applicable push from the [Push Scenarios by Mobile App](#) list and click the [Change](#) button.
5. Each push scenario is assigned to the SAP Mobile Platform, or on-premise trigger, by default. Change the [Active](#) outbound trigger for the push scenario to the `_HCP` outbound trigger and click [Save](#). Repeat this process for each push scenario you use in your application.
6. If you are on SMFND 620_700 Add-On Support Package SPO2 or lower, apply the following SAP Release Note:
 - o [2374370](#): Support Custom Header Parameters in Standard Outbound Trigger for Push

7.13.3 Push Scenarios for Operations and Suboperations Overview

The default implementation of SAP Work Manager includes functionality for server push behavior for emergency work orders. That functionality allows the back end to push emergency work orders to mobile clients before a user manually transmits. Push functionality includes support for complex work orders with multiple technicians assigned to different operations for the emergency work order. The default behavior of push functionality pushes the work order to all technicians with operations assigned to them. The default behavior occurs both when the emergency work order is approved, and if any operation within the work order changes.

You may want to only push an emergency work order to a technician if a change occurs to an operation to which the technician is assigned. For example, an emergency work order is already pushed to the technician, then an operation is changed, so an emergency push takes place. The same optional behavior is also available for suboperations.

To change the default push behavior, modify the following elements within the Mobile Add-On for ERP ConfigPanel:

- `/SYCLO/PM_EFI_AUFNR_EX_INCL`: EFI trigger
- `SWM65_EMERGENCY_WORKORDER_PUSH`: Push scenario
- `SWM65_EMERGENCY_WORKORDER_OPERATION_PUSH/`,
`SWM65_EMERGENCY_WORKORDER_SUBOPER_PUSH`: Push scenario
- `/SYCLO/PM_DOPUSHWORKORDER_GET`: BAPI wrapper

Modifying the following elements only affect the application if the overall server push functionality is enabled for the implementation.

There is a corresponding push definition within the Agency application project for the SAP Work Manager application. Because the push scenarios modified for the behavior described in this topic control the specific technicians that receive specific work orders, the push definition needs no modification in Agency to support these changes.

Note

Push is supported out of box for work order assignment types 1-3. For information on supporting other assignment types, see [2547564](#): SAP Work Manager: Push Support for Assignment Types 4-8 and A (MRS). You can only configure push for one assignment type at a time.

See the topic [Work Order Assignment Type Options \[page 110\]](#) for details.

7.13.3.1 Configuring Operation or Suboperation Push Scenarios

Prerequisites

Address the following items before performing this procedure:

- If you have not enabled the overall server push feature, the changes made in this procedure have no effect on the behavior of the mobile application. For more information, see the [Configuring Push \[page 124\]](#) topic.
- You must have access to the Mobile Add-On for ERP ConfigPanel and have permissions to change the configuration settings of the elements within it.

Context

The following procedure describes the steps necessary to change the push behavior of the emergency work order of the SAP Work Manager application. When the procedure is complete, technicians only receive updates to emergency work orders through a server push if the operation or suboperation to which the technician is assigned is modified. When an emergency work order is initially pushed to the SAP Work Manager clients, all technicians with one or more operations or suboperations assigned to them receive the work order. This aspect of the push behavior is unaffected by this modification. It is only when subsequent changes are made to an operation or suboperation for the work order where the difference in the push behavior is exhibited.

Procedure

1. Select the [EFI Assignment](#) link from the main page of the ConfigPanel.
2. From the [Enhancement Implementation Includes](#) list, expand the [/SYCLO/PM](#) node. Select the [/SYCLO/PM_EFI_AUFNR_EX_INCL](#) item. Select the [Assignment](#) tab.
3. From the list of assignments, select the [<APP>_WORK_ORDER_OPERATION_PUSH](#) exchange object in the list. Click the [Change](#) button. Check the [Active Flag](#) box and click the [Save](#) button.
4. On the same line, uncheck the [Active Flag](#) box for [<APP>_WORK_ORDER_PUSH](#).
5. Return to the ConfigPanel main page and select the [Push Scenario Definition](#) link. Select your application in the [<Mobile Application Filter>](#) field.
6. Select [<APP>_EMERGENCY_WORKORDER_PUSH](#) from the [Push Scenarios by Mobile App](#) list. Click the [Change](#) button.
7. In the [Activation](#) section, change the [Active Flag](#) setting by unchecking the box to disable the push scenario. Click [Save](#) to apply the modification.
8. In the list of push scenarios on the left of the page, select either the [<APP>_EMERGENCY_WORKORDER_OPERATION_PUSH](#) or [<APP>_EMERGENCY_WORKORDER_SUBOPER_PUSH](#) scenario, depending on the desired behavior. Click the [Change](#) button.
9. In the [Activation](#) section, check the [Active Flag](#) checkbox to enable the selected push scenario. Click [Save](#) to apply this modification.
10. Return to the ConfigPanel main page and select the [BAPI Wrapper Configuration](#) link. In the [BAPI Wrapper List](#) on the left of this page, expand the node [/SMERP/BAPI_PM_WO_PUSH](#). Under this node, select the BAPI wrapper [/SMERP/PM_DOPUSHWORKORDER_GET](#).
11. Select the [Assignment](#) tab to list the MDOs to which the BAPI wrapper is currently assigned.

Results

With the completion of this procedure, emergency work orders already received by technicians and stored on their client devices are not pushed again when changes occur unless the operation or suboperation modified is assigned to them.

Next Steps

Thoroughly test the push behavior after you complete the procedure. Verify that emergency work orders are initially pushed to all technicians to which they are assigned. Subsequent changes to either an operation or suboperation for a work order are only pushed down to the technicians to whom the operation or suboperation is assigned. Once testing is successful, migrate the modifications to the production system according to migration processes in place at the implementation site.

7.14 Integrating GIS with the SAP Work Manager Application

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

The SAP Work Manager and SAP CRM Service Manager applications have custom map controls with GIS functionality implemented using SAP EAM and service mobile app SDK and Open UI extensions. Their functionality is configured through the Agency application definitions. Both mobile applications are delivered with predefined mobile parameters that are Esri-specific. However, you have the option to point to any GIS supplier you choose.

Back End Configuration through the ConfigPanel

Find details of basic configuration for Esri integration in the ConfigPanel in the topic, .

Enabling and Disabling GIS Parameters using MDO Configuration

You can enable and disable the GIS parameters through the [Mobile Data Object Configuration](#) page of the ConfigPanel:

1. Select [Mobile Data Object Configuration](#) from the Home page of the ConfigPanel.
2. Under [CT - Complex Table](#), select the MDO `SWM65__GIS_PARAMETER_LIST`.
3. Select the [Data Filter](#) tab. Navigate to the `GIS_PARAMETERS` filter under the `GET` method.
4. Click the [Change](#) button.

5. Highlight the rule you want to change in the *Rule list*, for example:
 - *Parameter Group*: GIS_COMMON
 - *Param. Name*: PREF_GIS_OFFLINE_ENABLED
 - *Parameter Value*: TRUE
 - *Active Flag*: Active / Inactive
6. Make your changes, then click *Enter* to record the change.
7. *Save* your changes.
8. If applicable, enter the *Transport Request* to include the changes.

GIS with SAP Work Manager

The following BAPIs and MDOs are specific to using GIS with SAP Work Manager:

BAPIs:

- **Fetch**: /SMFND/GIS_OBJECT_GEODATA_GET
- **Add**: /SMFND/GIS_OBJECT_GEODATA_CRT
- **Edit**: /SMFND/GIS_OBJECT_GEODATA_UPD
- **Fetch**: /SMFND/CORE_CTGISPARAMLIST_GET

MDOs:

- **CT - Complex Table MDO**: SWM65_GIS_PARAMETER_LIST
- **DO - Standard Data Object MDO**: SWM65_GIS_OBJECT_GEOMETRY_GENERIC

7.15 Working with Inbound Transactions

Inbound transaction management is critical to ensure that data updates to the backend SAP system are posted properly and data integrity is preserved.

Inbound transaction management provides the functionality of asynchronous transaction processing. This functionality allows mobile client transactions to be uploaded and checked into inbound transaction queues in the SAP system. Transactions parked in the inbound transaction queues are processed asynchronously at a later time.

Inbound transaction management can be used for regular transaction processing or transaction error handling. Inbound transaction management can be enabled based on object type. A hybrid mode of mixing asynchronous transaction processing with synchronous transaction processing based on object types is also supported.

By default, inbound transaction management is turned off in SAP Work Manager. Mobile client transactions are uploaded and posted to the SAP ERP system synchronously.

Key Concepts

To understand inbound transaction management, learn the following key concepts:

- **Synchronous Transaction Management:** Default processing mode for all transactions. With synchronous processing, there is no inbound queue. In synchronous mode, the inbound transaction is processed and posted into the SAP system immediately. The back-end system response is sent back to the calling program (SAP Mobile Platform server) in real time.
- **Asynchronous Transaction Management:** Asynchronous transaction processing is optional and is disabled by default. Asynchronous transaction processing is managed by an inbound transaction queue. In this mode, a transaction checks in and parks in an inbound transaction queue. Processing of the transaction is delayed until a later time. Administrators can inspect, edit, and process transactions in the inbound queue on the back end. Activate inbound transaction queue management at the mobile application level.
- **Inbound Queue Configuration Object:** An inbound queue is the essential component in asynchronous inbound transaction management. An inbound queue defines what transaction can be checked into the queue, and how the transaction can be processed and edited. An inbound queue is defined as a configuration object using the ConfigPanel customizing tool. Inbound queues are mobile application-specific and you can define multiple inbound queues for each application. You can only designate one inbound queue as the primary inbound queue for a mobile application. The primary inbound queue can accept any kind of inbound transactions. Regular inbound queues typically restrict inbound transactions to specific types of transactions. For example, a restriction is based on object type, such as work order or service order queue, or notification queue.
- **Inbound Queue Assignment:** You can assign Inbound transactions to an inbound queue based on the technical object or interface that generated the transaction, for example, a BAPI wrapper. You can also base the The inbound queue assignment on transaction attributes such object type, or object group.
- **Inbound Queue Handler:** An inbound queue handler is the queue manager assigned to an inbound queue. The inbound queue handler provides the technical implementation that manages the standard inbound queue activities. Examples of activities include transaction input data parsing, transaction check-in process, transaction check-out and edit process, transaction cancellation, transaction approval, transaction key-mapping, and transaction processing.
All inbound queue handlers must be subclassed from the queue handler base class `/SMFND/CL_IBQ_INBQUEUE_MGR_ABS`. A standard queue manager, `/SMFND/CL_INB_Q_MGR_STD`, is provided to support standard inbound queue activities. Develop specialized inbound queue handlers to handle object type specific inbound queue requirements, such as key-mapping, approval, and processing.
- **Inbound Queue Check-In Process:** Multiple transaction check-in modes are supported. Check-in modes are described in the following bullet points as part of the *General Data* tab description.
- **Inbound Queue Approval Workflow:** You can activate an inbound queue approval workflow for an inbound queue. If an approval workflow is activated, you first review and approve the transactions checked into the queue before they are processed.
- **Inbound Queue Authorization:** Authorization check is important for the inbound queue process. Some common inbound queue activities that might require additional authorization include transaction check-out and edit, transaction approval, transaction cancellation, transaction processing, and transaction deletion.
- **Inbound Transaction Lifecycle:** Full lifecycle support is provided to transactions checked into an inbound transaction queue. You can assign an expiration date to transactions, then expired transactions are not processed. The inbound transaction lifecycle includes the following states:
 - NEW: The transaction is initially checked in without processing
 - REVIEW: An inbound queue approval workflow is required for the transaction

- ERROR: The transaction was processed and resulted in error
- READY: The transaction data was modified and is ready to be reprocessed
- CANCEL: The transaction was canceled and can no longer be processed. Purge the transaction.
- COMPLETE: The transaction has been processed successfully. Purge the transaction.
- **Inbound Transaction Processing:** Processing of transactions in an inbound queue supports transaction dependency (in which a defined condition must be checked and satisfied before processing), key-mapping, error handling, reprocessing, and conflict resolution. In addition, transaction process history is recorded.
- **Inbound Queue Monitor and Transaction Inspector:** The inbound monitor is the central tool provided to monitor and manage any inbound transaction in an inbound queue. The transaction inspector tool is the tool used to manage a specific inbound transaction and is fully integrated into the inbound monitor. Using the central tool, an authorized user can view transaction data, check-out and edit transaction data, and approve, cancel, process, or delete a transaction.
- **Primary Queue:** Out of box, the primary queue is blank. To set one queue as a primary queue, where all transactions are captured in this primary queue, select a queue from the standard queues. When you select one queue as a primary queue, it means that all transactions have the same properties and are used for simple mappings. However, further customization is needed in the object type assignment of the primary type queue for BAPI wrapper settings, cross references, key mappings, and other settings that are not detailed in this guide.

The screenshot shows the 'Mobile Application (Display Mode)' configuration page. The 'Inbound Transaction Management' section is highlighted with a red box, containing the following fields:

- Inbound Transaction Active:
- Standard Inbound Queue Id:

Other sections visible include:

- Basic Data:** Mobile Application: SAP_WORK_MANAGER_65, Type: Standard Application, Description: SAP Work Manager 6.5, Release: 6.5
- User Management Setting:** Disable Automatic User Creation:
- Server Management Setting:** Disable Automatic Server Registration:
- Lifecycle Management:** Application Blocked: Effective Date: 11/30/2017 Time: 10:00:00
- xChange Setting:** Disable Change Detection:
- Multi Backend Setting:** Multi Backend Enabled: System Role: Host
- Administrative Info:** Created By, Last Changed By, Creation Time Stamp, Changed Time Stamp

Inbound Transaction Procedures

Use the following topics and procedures to configure inbound transaction in the SAP Work Manager application:

- [Enabling Inbound Transaction Management \[page 137\]](#)
- [Inbound Transactions - Client Object Locking Matrix \[page 138\]](#)
- [Inbound Queues in SAP Work Manager \[page 141\]](#)
- [Key Mapping Processing \[page 144\]](#)

- [Inbound Transaction Processing: Review and Approval Workflow \[page 146\]](#)
- [Developing a Custom Inbound Queue Handler \[page 147\]](#)

7.15.1 Enabling Inbound Transaction Management

To support asynchronous transaction processing, enable inbound transaction management in both the backend SAP system and the client application.

Procedure

1. Select the option *Mobile Application Configuration* from the home page of the ConfigPanel.
2. Select the mobile application that you are configuring from the *Defined Mobile Applications* list and click the *Change* button.
3. Enable inbound transaction management in the back-end system processing. From the *General* tab, enable the *Inbound Transaction Active* checkbox under the *Inbound Transaction Management* section.

The screenshot shows the configuration page for a mobile application in the SAP GUI. The 'General' tab is selected. The 'Basic Data' section shows the application name as 'SAP_WORK_MANAGER_65' and its type as 'Standard Application'. The 'Inbound Transaction Management' section has the 'Inbound Transaction Active' checkbox checked. The 'Administrative Info' section shows the application was created by JEGANESH on 07/13/2017 and last changed by CARTERCL on 04/13/2018.

4. In the SAP GUI, use transaction code SU3 and add the parameter `/SMFND/INB_TRANS` to the *Parameter* tab to enable inbound transaction for each user ID in the back-end system.
5. Go back to the ConfigPanel main screen and click *Inbound Transaction Queue Definition*. Select the parameter in the application where you enabled inbound transaction in the previous step and check the *Require User Parameter* box. Ensure that the *Activation Token Value* field is set to *ON*.

Inbound Transaction Queue Definition Detail (Display Mode)

Create Copy Delete Change

General Data Object Type Assignment Security Settings

Basic Info

Inbound Queue Id: SWM65_FUNCLOCATION_QUEUE Inb. Queue Desc.: Funcional Location Queue
 Mobile Application: SAP Work Manager 6.5

Queue Manager Handler Info

Inbound Queue Handler: /SMFND/CL_INB_Q_MGR_STD : Inbound Transaction Queue: Standard Queue Manager
 Check-In Mode: Error Processing Only
 Require User Parameter: Parameter Id: /SMFND/INB_TRANS Activation Token Value: ON

6. Enable inbound transaction support in the client application logic. From the ConfigPanel home page, click the *Mobile Application Configuration* link, *Client Globals* tab. Find the following client globals and set their values to either *Y* or keep blank:

Global Group	Global Name	Global Value
INBOUND_TRANSACTION_QUEUE	DisplayErrorMessage	Y
INBOUND_TRANSACTION_QUEUE	FetchErrorMessage	Y
INBOUND_TRANSACTION_QUEUE	Save	Not Applicable

i Note
 Reflects the *Inbound Transaction Active* flag on the *General* tab.

Client Globals

Client Global List

Import/Export Search: INBOUND_TRANSACTION_QUEUE

RecNo	Global Group	Global Name	Global Value	Scope	Dep. RecNo	Active	No Change	Comment
0000000940	ITEMTASKSUCCESS	Lock Notification	N	User	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000939	ITEMTASKSTART	Lock Task	Y	User	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000938	ITEMTASKSTART	Lock Notification	N	User	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000937	ITEMTASKCOMPLETE	Lock Task	Y	User	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000936	ITEMTASKCOMPLETE	Lock Notification	N	User	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000853	INBOUND_TRANSACTION_QUEUE	Save		Application	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Reflects "Inbound Transaction Active" f...
0000000935	INBOUND_TRANSACTION_QUEUE	FetchErrorMessage	Y	User	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000934	INBOUND_TRANSACTION_QUEUE	DisplayErrorMessage	Y	User	0000000000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
0000000829	HOURS	H00	00	Application	0000000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	HOURS defines static data and is used...
0000000830	HOURS	H01	01	Application	0000000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	HOURS defines static data and is used...

7.15.2 Inbound Transactions - Client Object Locking Matrix

If inbound transaction management is enabled, special object locking logics are applied to client objects, because client transactions are uploaded and parked in the inbound transaction queues in the SAP system, and not posted against SAP business object data.

A global parameter with a parameter group of *INBOUND_TRANSACTION_QUEUE* and a parameter name of *Save* are defined in the ConfigPanel to turn on and off the transaction inbound management feature. By default, the global is set to *N*.

A global parameter with a parameter group of *INBOUND_TRANSACTION_QUEUE* and a parameter name of *DisplayErrorMessage* are defined in the ConfigPanel to turn on and off the transaction inbound management status fetch BAPI. By default, the global is set to *Y*. The global is *Y* by default to eliminate another property to configure if you want the transaction inbound management feature active on your system. Therefore, the fetch BAPI should only run when this parameter is enabled.

The global parameters in the following tables are defined to turn on and off the locking of the object as well as the object parent. The client locks the object based on the defined global parameter, if the value is *Y*, and if at least one error exists on the object.

SAP Work Manager Transactions

SAP Work Manager Transaction (Global Group)	Object (Global Name)	Lock (Global Value)
Work Order Edit	Work order	Y
Component Issue	Component	Y
	Work order	N
Document Add	Document	N
	Work order	N
	Notification	N
	Equipment	N
	Functional location	N
Characteristic Value	Characteristic value	Y
	Equipment	N
	Functional location	N
	Work order	N
	Notification	N
Measuring Point	Measuring point	Y
	Equipment	N
	Functional location	N
	Work order	N
	Notification	N
CATS Record Add/Edit	CATS	Y
Labor Add/Edit	Labor	Y

**SAP Work Manager Transaction
(Global Group)**

SAP Work Manager Transaction (Global Group)	Object (Global Name)	Lock (Global Value)
Notification Edit	Notification	Y
Notification Status	Notification	Y
Notification Task Status	Task	Y
	Notification	Y
Notification Item Task Status	Task	Y
	Notification	Y
Equipment Edit	Equipment	Y
Install Equipment	Equipment	Y
	Functional Location	Y
Uninstall Equipment	Equipment	Y
	Functional Location	Y
Functional Location Edit	Functional Location	Y
GIS Update	GIS Update	Y
	Equipment	Y
	Functional location	Y
	Work order	Y
	Notification	Y

Customer Service Transactions

**Customer Service Transaction (Global
Group)**

Customer Service Transaction (Global Group)	Object (Global Name)	Lock (Global Value)
Address Edit	Address	Y
	Business partner	Y

Meter Management Transactions

Meter Management Transaction (Global Group)	Object (Global Name)	Lock (Global Value)
Meter Install	Meter	Y
Meter Replace	Work order	Y
Meter Remove		
Meter Repair		
Aperiodic Meter		
Meter Disconnect		
Meter Reconnect		
Periodic Meter	Meter	Y

7.15.3 Inbound Queues in SAP Work Manager

To process a client transaction asynchronously, a client transaction is assigned to an inbound transaction queue in the SAP system and checked into the queue during client transmit.

In SAP Work Manager, inbound transaction queues are predefined for the following object types:

Object Type	Inbound Queue
Document attachment	SWM65_ATTACHMENT_QUEUE
Classification	SWM65_CLASSIFICATION_QUEUE
Equipment	SWM65_EQUIPMENT_QUEUE
Functional location	SWM65_FUNCLOCATION_QUEUE
GIS location	SWM65_GEODATA_QUEUE
Goods movement	SWM65_GOODSMVMT_QUEUE
Timesheet	SWM65_HR_CATS_TIMESHEET_QUEUE
Measurement document	SWM65_MEASUREMENT_DOCUMENT_QUEUE
Notification	SWM65_NOTIFICATION_QUEUE
Purchase order	SWM65_PURCHASEORDER_QUEUE

Object Type	Inbound Queue
Work order confirmation	SWM65_WORKORDER_CONFIRMATION_QUEUE
Work order	SWM65_WORKORDER_QUEUE

BAPI Wrappers and Inbound Transaction Support

Client transactions are transformed into BAPI wrapper calls. To support inbound transactions, assign a BAPI wrapper to an MDO method of create, update, or delete. In addition, the BAPI wrapper must implement the following input and output table parameters:

- Input Table Parameter: *IT_TRANSACTION_REQ_INFO*, with data type */SMFND/IBQ_TRANS_REQ_INFO_TAB*
- Output Table Parameter: *ET_TRANSACTION_REQ_STATUS*, with data type */SMFND/IBQ_TRANS_STA_TAB*

The general purpose BAPI wrapper */SMFND/INB_TRANS_REQ_STAT_GET* is provided to fetch the inbound transaction status for a mobile user.

Security Settings for the Inbound Transaction Queue

The back office users who manage and process the inbound transaction queues can access and modify client transaction data from various mobile users. Therefore, it is important that you define the appropriate authorization requirements for the back office users.

We recommend that you review and set the appropriate authorization checks for the inbound transaction queue, especially for the following activities:

- **Checking Out:** Allows the back office user to modify client transaction data before posting
- **Review and Approve:** When an approval workflow is activated, the authorization check allows the back office user to approve the transaction for posting

To define authorization requirements for inbound transaction queue, go to the ConfigPanel and select the *Inbound Transaction Queue Definition* option from the Home page. You can activate the authorization requirements from the *Security Settings* tab.

Inbound Transaction Queue Definition Detail (Change Mode)

General Data Object Type Assignment **Security Settings**

Queue Info

Inbound Queue Id: SWM65_FUNCLOCATION_QUEUE Mobile Application: SAP_WORK_MANAGER_65
 Queue Handler: /SMFND/CL_INB_Q_MGR_STD

Security Settings

Authorization Requirement

Add Delete

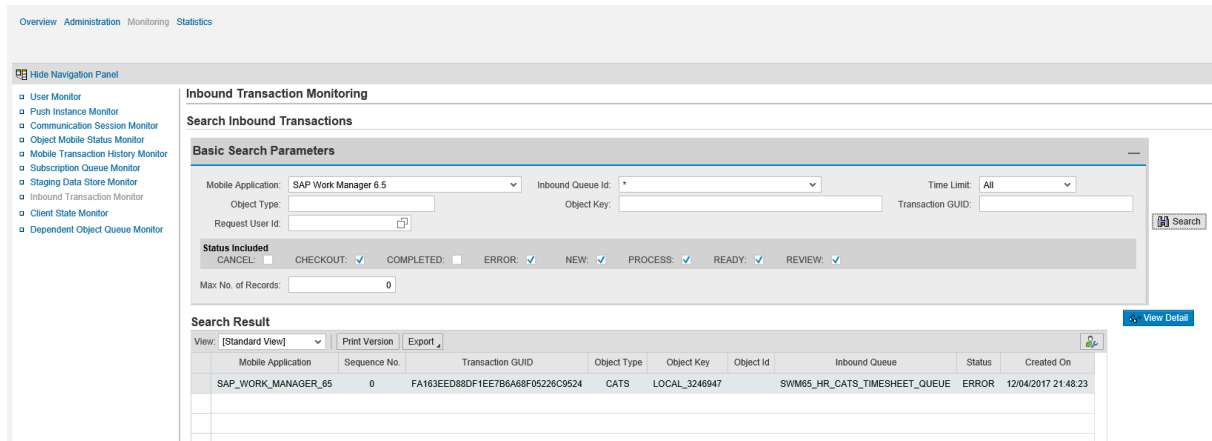
Check Type	User Role	Active
Review &... ▾	PURCHASING_MANAGER	<input type="checkbox"/>

Transaction Dependency for the Inbound Transaction Queue

Transaction dependency is the condition where there is a leading transaction, *A*, and at least one subsequent transaction, *B*, where the successful processing of the subsequent transaction *B* is dependent on the successful processing of the leading transaction *A*.

By default, determination of the existence of transaction dependency is handled by the client application, where transactions are originated. In addition, you can also use BAdI implementation to support transaction dependency determination during the inbound transaction check-in process. Use BAdI `/SMFND/IBQ_QUEUE_HANDLER_BADI` to implement transaction dependency.

You can view the transaction dependency information when inspecting an inbound transaction from the Admin Portal. To access the Admin Portal, run transaction `/n/syclo/admin`, then select **Monitoring** > **Inbound Transaction Monitor** .



7.15.4 Key Mapping Processing

When working with the inbound transaction queue, key mapping processing is required when a leading transaction creates a new object or journal entry in the SAP system that a subsequent transaction refers to.

For example, if a leading transaction, *A*, creates a new object or journal entry in the SAP system, and the subsequent transaction, *B*, refers to the SAP object ID or SAP journal entry created by leading transaction *A*, a key-mapping process is required during the processing of the transactions.

A local object ID, *LOCAL_X*, is assigned by the client application to identify the object being created by the leading transaction *A*. Subsequent transaction *B* references the object created by leading transaction *A* using the local object ID. During processing of the subsequent transaction *B*, the reference to the local object ID, *LOCAL_X*, is replaced with the actual SAP object ID generated after processing of the leading transaction.

Business case: An example of key mapping processing in SAP Work Manager is when the leading transaction creates a work order and the subsequent transaction creates an attachment for the new work order.

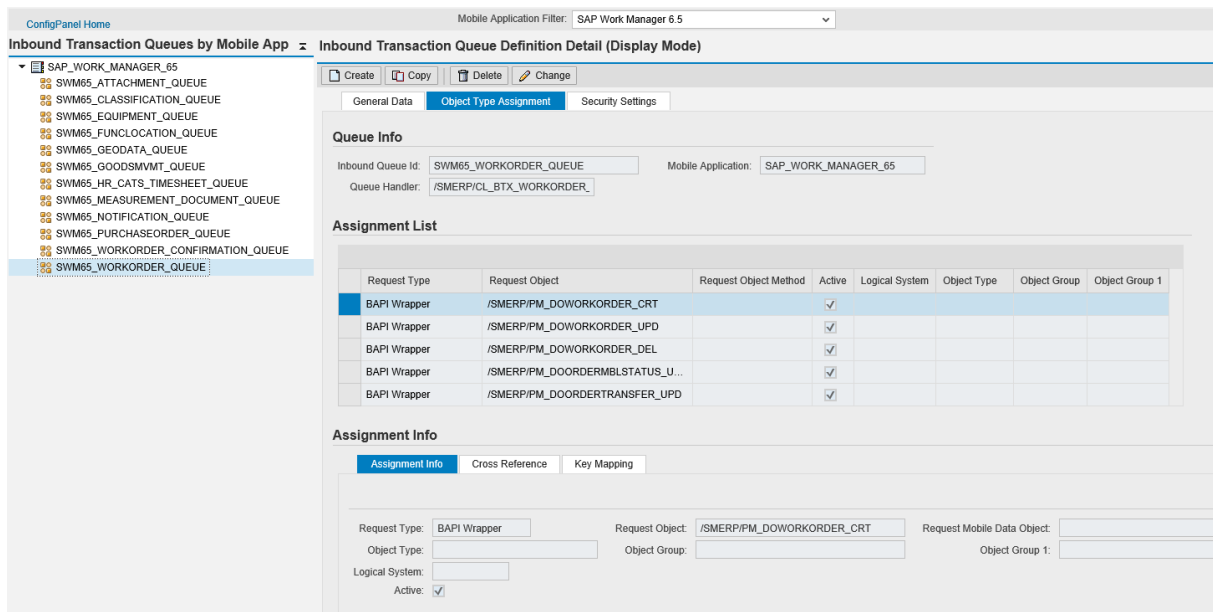
Configuration to support key mapping processing consists of the following:

- Define the cross reference setting for the leading transaction inbound queue
- Define the key mapping setting for the subsequent transaction inbound queue

Define Cross Reference Setting for Leading Transaction Inbound Queue

Defining the cross reference setting for the leading transaction inbound queue records the cross reference for the local object ID, *LOCAL_X*, and the SAP object ID.

In the example below, the cross reference setting is defined for the inbound queue **SWM65_WORKORDER_QUEUE** and BAPI wrapper */SMERP/PM_DOWORKORDER_CRT*. The cross-reference for the work order header GUID and line item GUID are recorded in the cross-reference table.



Following is a sample of a cross reference for a notification:

	MANDT	TRANSACTION_GUID	RECORD_ID	MOBILE_APP	OBJECT_TYPE	SRC_OBJECT_KEY
<input type="checkbox"/>	800	005056AC5ABD1EE68CCC06F296AF1DE9	1	SAP_WORK_MANAGER_64	NOTIFICATION_ID	10005265
<input type="checkbox"/>	800	005056AC5ABD1EE68CCC9FD205C97E9B	1	SAP_WORK_MANAGER_64	NOTIFICATION_ID	LOCAL_2016-06-14_14:17:40
<input type="checkbox"/>	800	005056AC5ABD1EE693B93BE98271A2A6	1	SAP_WORK_MANAGER_64	NOTIFICATION_ID	10005310

Following is a sample of a cross reference for a work order:

	MANDT	TRANSACTION_GUID	RECORD_ID	MOBILE_APP	OBJECT_TYPE	SRC_OBJECT_KEY
<input type="checkbox"/>	800	05B98157EDD54C1AE10000000A601AFA	1	SAP_WORK_MANAGER_64	ORDER_ID	000000000001
<input type="checkbox"/>	800	39A48157A2E49565E10000000A601AFA	1	SAP_WORK_MANAGER_64	ORDER_ID	000000000001
<input type="checkbox"/>	800	D282835789E45026E10000000A601AFA	1	SAP_WORK_MANAGER_64	ORDER_ID	000000000001
<input type="checkbox"/>	800	D5128157A7E4786BE10000000A601AFA	1	SAP_WORK_MANAGER_64	ORDER_ID	000000000001

Define Key Mapping Setting for Subsequent Transaction Inbound Queue

Defining the key mapping setting for the subsequent transaction queue instructs the system to replace the reference of the local object ID, LOCAL_X, in the subsequent transaction with the actual SAP object ID or journal entry. The reference is replaced using the cross reference record from the leading transaction before posting the subsequent transaction. Without key mapping, posting of the subsequent transaction results in failure because it is referring to an invalid LOCAL_X SAP object ID.

In the following example, a key mapping setting is defined for the inbound queue **SWM65_ATTACHMENT_QUEUE** and BAPI wrapper `/SMERP/CORE_DOBDSDOCUMENT_CRT`. The value of attribute *INSTID* for the attachment creation parameter *IS_BUSINESS_OBJECT* is replaced by the matching cross-reference record for *ORDER_ID* of the leading transaction work order creation.

Inbound Transaction Queues by Mobile App

- SAP_WORK_MANAGER_65
 - SWM65_ATTACHMENT_QUEUEUE
 - SWM65_CLASSIFICATION_QUEUEUE
 - SWM65_EQUIPMENT_QUEUEUE
 - SWM65_FUNCLOCATION_QUEUEUE
 - SWM65_GEODATA_QUEUEUE
 - SWM65_GOODSMVMT_QUEUEUE
 - SWM65_HR_CATS_TIMESHEET_QUEUEUE
 - SWM65_MEASUREMENT_DOCUMENT_QUEUEUE
 - SWM65_NOTIFICATION_QUEUEUE
 - SWM65_PURCHASEORDER_QUEUEUE
 - SWM65_WORKORDER_CONFIRMATION_QUEUEUE
 - SWM65_WORKORDER_QUEUEUE

Inbound Transaction Queue Definition Detail (Display Mode)

Create Copy Delete Change

General Data **Object Type Assignment** Security Settings

Queue Info

Inbound Queue Id: SWM65_ATTACHMENT_QUEUEUE Mobile Application: SAP_WORK_MANAGER_65
 Queue Handler: /SMFND/CL_INB_Q_MGR_STD

Assignment List

Request Type	Request Object	Request Object Method	Active	Logical System	Object Type	Object Group	Object Group 1
BAPI Wrapper	/SMERP/CORE_DOBDSDOCUMENT_CRT		<input checked="" type="checkbox"/>				
BAPI Wrapper	/SMERP/CORE_DODMSDOCUMENT_CRT		<input checked="" type="checkbox"/>				
BAPI Wrapper	/SMERP/CS_DOBDSDOCUMENT_CRT		<input checked="" type="checkbox"/>				

Assignment Info

Assignment Info Cross Reference **Key Mapping**

Key Mapping Info

Key Mapping Setting List

Active	Mapping Type	Reference - Object Type	Target Key Reference Path	Target Key - Attribute
<input checked="" type="checkbox"/>	Replace	ORDER_ID	IS_BUSINESS_OBJECT	INSTID
<input checked="" type="checkbox"/>	Replace	NOTIFICATION_ID	IS_BUSINESS_OBJECT	INSTID
<input checked="" type="checkbox"/>	Replace	EQUIPMENT_ID	IS_BUSINESS_OBJECT	INSTID
<input checked="" type="checkbox"/>	Replace	FUNC_LOC_ID	IS_BUSINESS_OBJECT	INSTID

7.15.5 Inbound Transaction Processing: Review and Approval Workflow

If there is a business requirement, a review and approval workflow is used during inbound transaction processing. If the workflow is enabled, approve the inbound transaction, and it is then processed.

To enable the review and approval workflow, set a required review in the ConfigPanel:

1. Select the *Inbound Transaction Queue Definition* link from the ConfigPanel Home page.
2. On the *General Data* tab, check the *Review Required* checkbox for the selected inbound transaction.

Inbound Transaction Queue Definition Detail (Change Mode)

General Data | Object Type Assignment | Security Settings

Basic Info

* Inbound Queue Id: SWM65_FUNCLOCATION_QUEUE * Inb. Queue Desc.: Functional Location Queue

* Mobile Application: SAP Work Manager 6.5

Queue Manager Handler Info

Inbound Queue Handler: /SMFND/CL_INB_Q_MGR_STD : Inbound Transaction Queue: Standard Queue Manager

Check-In Mode: Error Processing Only

Require User Parameter: Parameter Id: /SMFND/INB_TRANS Activation Token Value: ON

Dependency Checking

Dependency Scope: Application Only

Conflict Management

Conflict Mode: Last In Win

Approval Workflow

Review Required:

Activation

Active Flag:

Admin Info

Created By: JEGANESH Created Time Stamp: 07/13/2017 23:15:56

Last Changed By: GOFF Changed Time Stamp: 11/16/2017 20:18:47

Reviewing and Approving Inbound Transactions

Review and approve inbound transactions for processing from the Admin Portal.

7.15.6 Developing a Custom Inbound Queue Handler

If the standard inbound queue handler cannot support your business requirement, you can develop a custom inbound queue handler. Define an inbound queue handler as a subclass of `/SMFND/CL_INBQUEUE_MGR_ABS`.

7.16 Data Staging

By enabling data staging through the ConfigPanel, you can reduce the initial sync time for a very large data set to the mobile client.

Business case: assume that you are using an Agency application for SAP backend and during your go-live or user acceptance phase, you are expecting hundreds of users to concurrently connect to the SAP Mobile Platform and synch data for the first time. In normal scenarios, without data staging, you could see long wait

times for users trying to download complex table data. Additionally, there is a large load on the SAP back-end system. With data staging enabled, you can reduce the sync time significantly by prepopulating the complex table data ahead of time. The Agency mobile framework supports data staging from the SMFND 610_700 Add-On Support Package 04 and onwards.

Data staging is only recommended for the initial sync, provided data volumes are justified. Although there is no technical limitation of using it during later stages, data staging poses a different set of challenges to keep data staging tables up to date. You have the option to deactivate data staging after initial go-live or you can continue to use it.

i Note

Data staging is only supported for complex tables in standard product. By default, data staging is turned off. The use of data staging is only recommended for initial synch, dealing with static data to reduce time during initial synch and any errors due to data staging being out of synch. We recommend that you take advantage of the exchange mechanism framework for subsequent master and transactional data delta syncs.

General Configuration Steps for Activating Data Staging

The following steps provide general directions required for activating data staging. For specific tab or field details, see the appropriate topic in the chapter of this guide.

1. From the ConfigPanel Home screen, go to [Data Store Definition](#) > [Data Staging Framework Settings](#) and create a new data store definition.
Use the data store definition settings to define the data store settings, data storage, data agent assignment, schedules, and mapping information. A data store is a data storage that can provide data to a mobile data object. A data store can consist of multiple data segments, which are related data sets. A data store is populated by assigned data agents and schedules.
2. From the ConfigPanel Home screen, go to [Data Agent Definition](#) and create the data store agent definition.
Use the data store agent settings to define the agent type, data object including the BAPI wrapper, and MDO used by the data loading program, which is read during runtime. Then, assign the data agent to a data store.
3. From the ConfigPanel Home screen, go to [Mobile Data Object Configuration](#) and create a separate MDO for the data store.
You can either create a new MDO from scratch or you can copy an existing standard MDO object, changing the namespace and definition.
4. From the ConfigPanel Home screen, go to [BAPI Wrapper Configuration](#) and add the assignment for the data store MDO object at the BAPI wrapper level.
You can use the same BAPI for populating staging data and downloading data to the mobile client.
5. From the ConfigPanel Home screen, go to [Mobile Data Object Configuration](#) and on the Data Staging tab, activate data staging at the consuming MDO object level.
6. Execute the program SMFND/DSF_DRIVER_PROG to populate the data staging table.
7. To test data staging on the SAP side, execute the BAPI wrapper associated with the complex table.

7.17 Assigning Responsible Vehicles to Users

Assign responsible vehicles to users in the SAP Work Manager application through user attributes. User attributes are enabled by default in the ConfigPanel.

Following are the three core configuration steps to implement user attributes:

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

Create the User Attribute

Use the *User Attributes* tab of the *Mobile Application Configuration* section in the ConfigPanel to create new user attributes. To access the tab, go to **ConfigPanel** > *Mobile Application Settings* > *Mobile Application Configuration* and select the *User Attributes* tab.

The screenshot displays the 'User Attributes' configuration interface. At the top, there are tabs for 'General', 'Mobile Status Setting', 'Conversion Exit Setting', 'System Components', 'Parameters', 'Client Globals', and 'User Attributes'. The 'User Attributes' tab is active. Below the tabs, the 'Mobile Application Info' section contains fields for 'Mobile Application' (SAP_WORK_MANAGER_65), 'Release' (6.5), and 'Mobile App. Desc.' (SAP Work Manager 6.5). The 'User Attribute Settings' section features an 'Attribute List' table with the following data:

Attribute Name	Reference Structure Name	Reference Field Name	ValueSet Provider Class
TRUNK_LOCATION	/SMERP/MM_USER_LOCATION_STR		/SMERP/CL_MM_USRATTR_VALUESET

Below the table is the 'Attribute Detail' section with the following fields:

- * Attribute Name: TRUNK_LOCATION
- Description: Assigned trunk location
- Reference Structure Name: /SMERP/MM_USER_LOCATION_:
- Reference Field Name:
- ValueSet Provider Class: /SMERP/CL_MM_USRATTR_VALUESET - Value Set: User Attribute ValueSet Provider Plant/Location

Define the Data Consumption in the Mobile Data Object (MDO)

Once you have created, defined, and activated the user attributes, define how the data is consumed in the MDO. Navigate to **Mobile Data Object Configuration** > *Data Filter tab*. From the *Data Object Navigation Tree*, select **CT - Complex Table** > <APP>_USER_TRUNK_ASSIGNMENT. Expand the *Get Method - GET* filter tree until you see the *USER_TRUNK* filter.

i Note

Out of the box, *USER_TRUNK_STOCK* is enabled for the *USER_TRUNK_ASSIGNMENT* user rule.



The complex table associated with user attributes is *SWM65_USER_TRUNK_ASSIGNMENT*, and is enabled by default. To define how the user attribute data is to be consumed in the MDO, define the *USER_TRUNK* filter.

For the *<Table Rule Category>*, choose *Non-Static Table Rules*. Then select *Mobile User Attribute* in the *<Table Filter Rule Type>* field. The static data defined with the *Static Table Rules* selection is applicable to all users because it is application-specific, not user-specific. Instead, you are defining data that is specific to a user. That data comes from the Admin Portal.

Once you select the *Mobile User Attribute* filter category, check the *Active Flag* and populate the *Rule List*, if it is not already populated out of the box with the rules you need for your customer site.

i Note

The following MDOs are restricted by the user attribute feature:

- SWM65_TRUNK_STOCK
- SWM65_USER_TRUNK_ASSIGNMENT

Assign Attribute Values to Users

Once you have defined your user attributes in the ConfigPanel, define the values for the attributes and activate the attributes you plan to use. Navigate to the **Admin Portal** **Administration** **User Management** and select the *User Attributes* tab in the *Mobile User Detail* section of the screen.

Middleware User Management

Search Mobile Users

Basic Search Parameters

Mobile Application: SAP Work Manager 6.5 User: [] Search

Last Activity Time: All

Search Result

User Id	Mobile App	Server Serial No	Lock Flag	Created On
ACEPUSHUSR	SAP_WORK_MANAGER_65	SAP_WORK_MANAGER_65		01/08/2018 22:18:21
ACESERVUSR	SAP_WORK_MANAGER_65	SAP_WORK_MANAGER_65		11/07/2017 21:01:04
AGUSER12	SAP_WORK_MANAGER_65	SAP_WORK_MANAGER_65		11/30/2017 21:22:09
AGUSER13	SAP_WORK_MANAGER_65	SAP_WORK_MANAGER_65		11/30/2017 21:08:59
AGUSER2	SAP_WORK_MANAGER_65	SAP_WORK_MANAGER_65		11/29/2017 22:51:52

Mobile User Detail (Display Mode)

Change

Basic Info **User Attributes** Preference Info Cross Reference List Client Registration Info

Attribute Settings

Attribute Name	Enabled	Value

Attribute Detail

Attribute Name: []

Reference Structure: [] Reference Field Name: []

Active Flag:

Take the following steps to define and activate any applicable user attributes.

1. Select a user from the user list.
2. Click *Change* in the *Mobile User Detail* section.
3. Click *Add* in the *Attribute Settings* area.
4. Select an attribute from the *Attribute Name* menu. This list populates from the list of attributes you created in the ConfigPanel. Based on your selection, the *Reference Structure* field and *Reference Field Name* fields are populated with the appropriate data.

i Note

Out of the box, the only attribute you need to define is the *USER_LOCATION* assignment.

5. Optionally, select a *Plant* and *Storage Location* as appropriate.
6. Check the *Active Flag* checkbox to activate the selected user attribute.

i Note

You can go back and set more user attributes to *Active*. You also have the option to use the same user attribute with different values for the same user.

7. When you have finished defining and activating the user attributes, click *Save*.

8 Upgrade Considerations

When upgrading from a previous release of the SAP mobile application, there are general application tools that can help with the migration.

Mobile Application Data Migration Tool

The mobile application data migration tool `/SMFND/CORE_DATA_MIGRATE_PROG` allows you to copy the transactional data table data during an upgrade from a previous version of the application to the new application. This is done as a cutover task. You can copy the data from the following Transactional Data Tables:

`/SYCLO/MBL01` - Mobile Device Objects Status

`/SMFND/OID01` - Geospatial Mappings

Mobile Application Comparison Report Utility

The Mobile Application Comparison Report Utility allows you to compare configuration differences between two mobile applications, two mobile application BC sets, or a mobile application and a BC set. It also allows you to compare across clients. The transaction `/SMFND/MAPP_COMPARE` compares the mobile application configuration between two mobile applications. This is useful during the upgrade process because the report highlights the differences between the current mobile application and the new one activated from the BC set as part of the upgrade.

Disable Old Application



Once you have completed the upgrade, be sure to go back and disable the old application version. If you do not disable the old application, change detection will continue to run on both versions. For additional information, see the SAP Note #2116192 "FAQ - Mobile Add-On for SAP Performance Recommendations". <https://support.sap.com/notes>

Important Disclaimers and Legal Information

Hyperlinks

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

About the icons:

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

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

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

Gender-Related Language

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

© 2019 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.