Reference Data Management

for SAP Master Data Governance

Configuration

Version: 10.12.2025 RDM 2511

Content

1 Introduction

Reference Data Management (RDM) for SAP MDG provides the governance of reference data using pre-delivered reference data objects, user interfaces, workflows and the replication of the data to a local staging area in receiving SAP systems. If you encounter any problems with this guide do not hesitate to get in contact with us. Please use support@itego.de (subject: "Configuration Guide RDM: <topic>").

2 Prerequisites

RDM is built as an Add-On to SAP Master Data Governance (MDG) which means that it needs to be installed based on SAP MDG. See the RDM Installation Guide for more details.

SAP MDG for Custom Objects (MDG-CO) needs to be activated and certain configuration steps need to be performed. See section 3.1 "Activate Business Functions".

3 Configuration for Reference Data Governance

3.1 Activate Business Functions

Before you activate business functions, ensure that you have the administration authorization for MDG. The required authorization objects are delivered with the authorization role SAP_MDG_ADMIN. In transaction PFCG, we recommend to create a copy of this role and to assign the relevant authorizations. For authorization object USMD_DM Data Model you need to assign the value for field "USMD_MODEL": "I1" and the values for "ACTVT" (e.g. 01: Create or generate and 02: Change).

In transaction SFW5 "Activate Business Functions" activate the following business functions for MDG-CO:

- MDG_FOUNDATION
- MDG_FOUNDATION_2
- MDG_FOUNDATION_3
- MDG_FOUNDATION_4
- MDG_FOUNDATION_5
- MDG_FOUNDATION_6

And activate the following business function for RDM if you want to use the RDM Product Hierarchy:

- /ITR/FOUNDATION_01

Also activate the Web Dynpro Applications for MDG-CO (see Configuration Guide for SAP MDG Custom Objects 8.0 -> "Services to be activated for MDG Web Dynpro Applications"). This document can be found here http://help.sap.com/mdg (MDG based on SAP ERP -> Application Help -> Version 8.0 -> Configuration of SAP MDG -> Config. of SAP MDG Central Governance -> Configuration of MDG for Custom Objects -> "Services to be activated for MDG Web Dynpro Applications").

Note: These business functions only need to be checked or activated on the SAP MDG system. No action necessary on the SAP Business systems which are usually the receivers of the data maintained on the SAP MDG system.

3.2 Activate Data Model I1

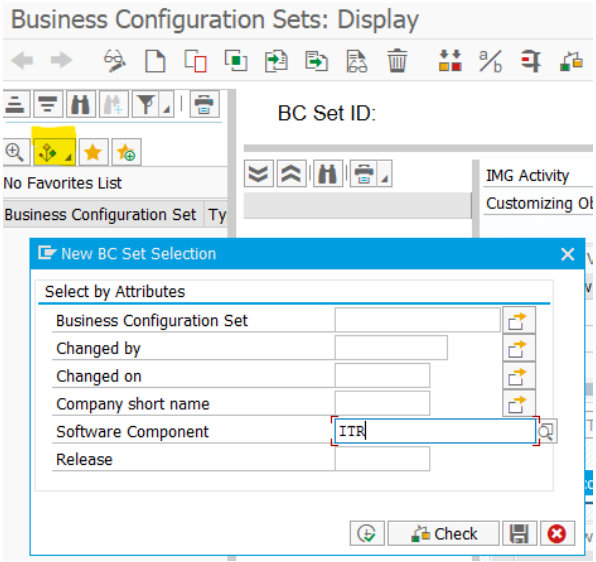
Check whether the data model I1 has been activated in transaction MDGIMG: General Settings -> Data Modeling -> Edit Data Model. If it has not been activated, select Data Model "I1" and click on  "Activate".

3.3 Activate Business Configuration Sets

The following Business Configuration Sets (BC Sets) might have to be imported on the SAP MDG system. Use transaction SCPR20 "Activate Business Sets" to activate them shown in the sequence below.

Please be aware that, as with any BC Set, you should check for conflicts before activating them. If there are conflicts, please investigate if you would like to activate anyways, partially or not, as entries in your SAP MDG implementation might be affected. In doubt, please create a backup including the affected configuration tables in a transport, which can be used to restore your settings later if necessary. Please consider to clarify any remaining question by contacting support@itego.de (subject: "BC Set Usage")

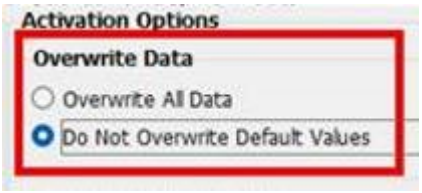
You can use transaction SCPR3 "Display and maintain BC Sets" to investigate the content of the BC Sets. Please use "Select by Attributes":



And search for Software Components ITR, ITO and ITG (on the MDG System) or ITO and ITG (on an SAP Business System).

Note: Starting with RDM 2407 the latest available BC Set includes the complete content from earlier versions. This means that for the installation of newer versions than RDM 2403 only the latest available BC Set (highest number <n>) needs to be activated. Please also be aware that new BC Sets (for the Fiori Launchpad) have been delivered which need to be activated as well. Please make sure that BC Sets are completely activated, in doubt repeat the activation step and if no complete activation is possible, contact support@itego.de (subject: "BC Set Activation").

Important Note for Upgrades: For the upgrade and the activation of new features, please make sure that "Do Not Overwrite Default Values" (see below) is used. If in doubt, please check with support@itego.de for instructions.



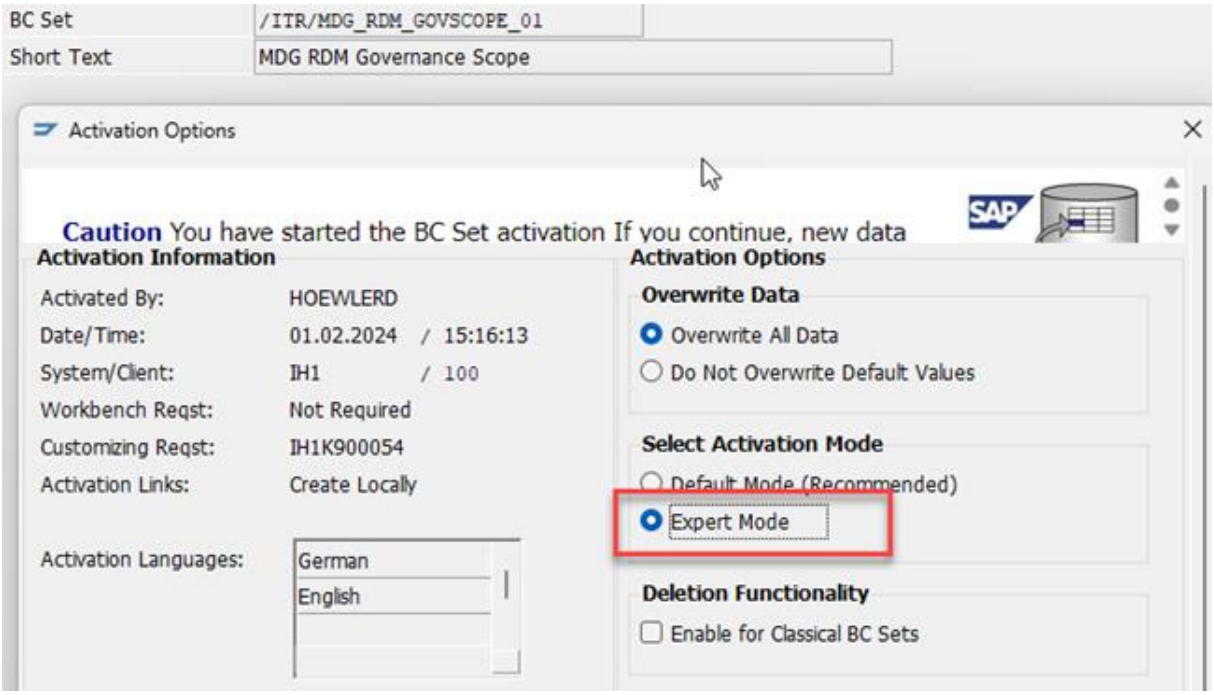
3.3.1 SAP MDG: BC Sets

3.3.1.1 Software Component ITR

The following BC Sets need to be activated on the MDG system:

- /ITR/MDG_RDM_BASIC_FRMWRK_<n> MDG RDM Framework Basic Settings
- /ITR/MDG_RDM_BUSACT_<n> MDG RDM Business Activities
- /ITR/MDG_RDM_DRF_BUS_ALT_<n> MDG RDM Replication Business Alternative
- /ITR/MDG_RDM_GOVSCOPE_<n> MDG RDM Governance Scope
- /ITR/RDM_SEARCH_HELP_<n> MDG RDM Search Help Selection
- /ITR/FLP_SEM_OBJC_<n> Fiori Launchpad Semantic Objects
- /ITR/FLP_SPACES_PAGES_<n> Fiori Launchpad Spaces and Pages

Note: /ITR/MDG_RDM_GOVSCOPE_<n> needs to be activated in "Expert Mode" which might need additional authorizations.



Besides the BC Sets listed above, the following BC Sets deliver Change Request types and Workflow configurations (**check the note below before activating**):

- /ITR/MDG_RDM_CREQUEST_<n> MDG RDM Change Request Types
- /ITR/MDG_RDM_WORKFLOW_<n> MDG RDM Workflow

Important Note: It is necessary to activate /ITR/MDG_RDM_WORKFLOW_01 as well, if you want to use the delivered change request types. Please handle with care, as this BC Set delivers Step Types, Actions and Change Request Status that might already have been used in your system. These are:

- Step Types: A, E, H, P
- Actions: I1
- CR Status: 51, 53, 54, 55, 98, 99

Please make sure that you at least create a backup of the current settings and align them with the setting from the BC Sets after activation. In doubt **do not** activate this BC Set. If you do not activate it, you either have to use your own change request types or you need to adjust the delivered change request types.

For Data Transfer please activate these BC Sets:

- /ITR/MDG_RDM_DT_<n> MDG RDM Data Transfer Settings

In order to get some predefined File Upload Variants please also consider to activate the following BC Sets (please be aware that not all possible use cases are covered):

- /ITR/MDG_RDM_UPLOAD_VAR_<n> Upload Variants for RDM Objects

Note: In earlier versions of RDM the BC Set /ITR/MDG_RDM_BRF_DRF_<n> has been used to deliver templates for the configuration of the Data Replication. Do not activate these but review chapter 3.10 Set Up Data Replication for additional information.

3.3.2 SAP Business Systems: BC Sets

Note: It is quite common that MDG Systems have more than one client. One client (e.g. 100) could be used for the MDG system itself and another client or clients (e.g. 400 and 401) could be used to serve as "(Test) Business Systems". In this case activate also the BC Sets mentioned for Software Component ITO on the MDG system clients that you actually use as "(Test) Business System".

3.3.2.1 Software Component ITO

The following BC Set has to be imported to all SAP business systems:

- /ITR/RDM_STAGING_<n> MDG RDM Local Staging Area

The following BC Sets should be imported to the SAP business system, which is used for the extraction of reference data (used for the initial load of the RDM system)

- /ITR/RDM_MDMGX_<n> MDG RDM Data Export

The following BC Sets have to be imported to all SAP business systems which will be integrated to SAP Solution Manager ChaRM (not required for XLD):

- /ITR/MDG_RDM_SOL_MAN_<n> MDG RDM Solution Manager Integration

3.4 Configure User Roles and Authorizations

3.4.1 Reference Data Governance – MDG

The following roles are delivered for Reference Data Governance on the MDG system:

- /ITR/ITEGO_MDG_RDM_MENU_<n> NWBC Menu
- /ITR/ITEGO_MDG_RDM_DISP_<n> Display Role
- /ITR/ITEGO_MDG_RDM_REQ_<n> Requester
- /ITR/ITEGO_MDG_RDM_SPEC_<n> Data Specialist
- /ITR/ITEGO_MDG_RDM_STEW_<n> Data Steward

These Requester, Data Specialist and Data Steward Roles are reused in the BRF Workflow definitions for the Change Requests described in section 3.9 "Configure Rule Based Workflow". It is recommended to copy the roles to your own namespace and replace the roles in the Workflow definitions.

3.4.2 Reference Data Governance – SAP Business System

The following roles are delivered for Reference Data Governance on the SAP business systems:

- /ITR/ITEGO_LSA_MENU Local Staging Area Menu
- /ITR/MDG_LSA_ADMIN Local Staging Area Administration
- /ITR/MDG_STAGING_<n> Local Staging Area
- /ITR/MDG_LSA_SNAPSHOT_MNG_<n> Snapshot Management

Make sure to adjust the authorization objects of your roles to define which roles can access and work with which reference data object types. The delivered roles should be excluded from the authorization profiles for other roles.

After the adjustment of the roles, assign your users to the roles and make sure that data model "I1" is assigned to the user profile parameter R_FMDM_MODEL "SAP Master Data Governance"

The following role is delivered for the Master Data Framework on the MDG system and enables the user to maintain Business Rule configurations:

- /ITU/MDF_RF_MNT_<n> MDF Rule Framework Maintenance

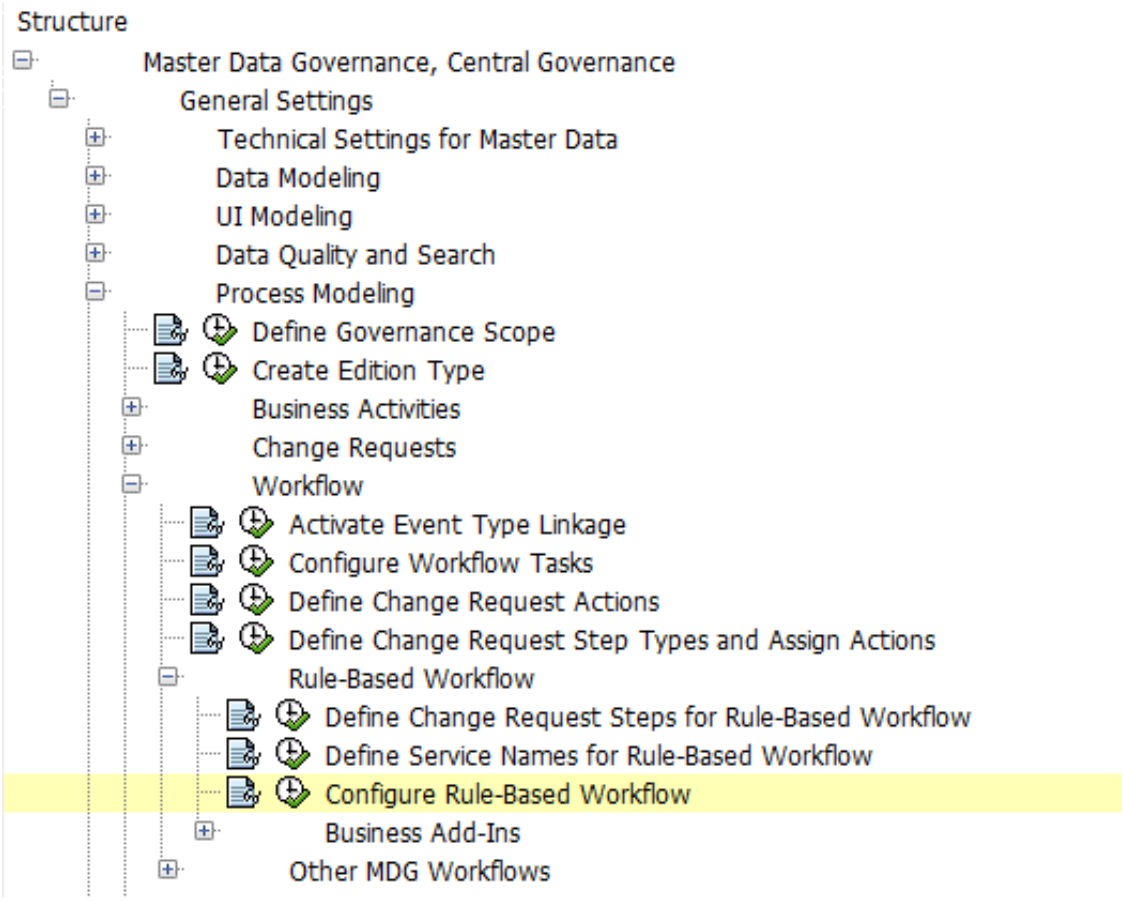
3.5 Copy Rule Based Workflows from Client 000

Rule based workflows delivered by RDM are delivered in client 000. In order to transfer these workflows to your operational client, you need to execute the following steps. Two alternatives are provided, the "automated copy" (which is recommended) or the "manual

copy". Please make sure that you make yourself familiar with the topic using the introduction provided below.

3.5.1 Introduction

Using transaction MDGIMG -> Process Modelling -> Workflow -> "Configure Rule-Based Workflow" provides access to the BRF+ (Business Rule Framework+) decision tables, that provide flexible processing of RDM change requests.



The BRF+ customizing used for this is delivered by SAP or SAP Partners like Itego to client 000 (using "C-tables"). This is how client specific data for BRF+ is delivered without overwriting data on target systems. For more information you can also check: BRFplus User Guide.

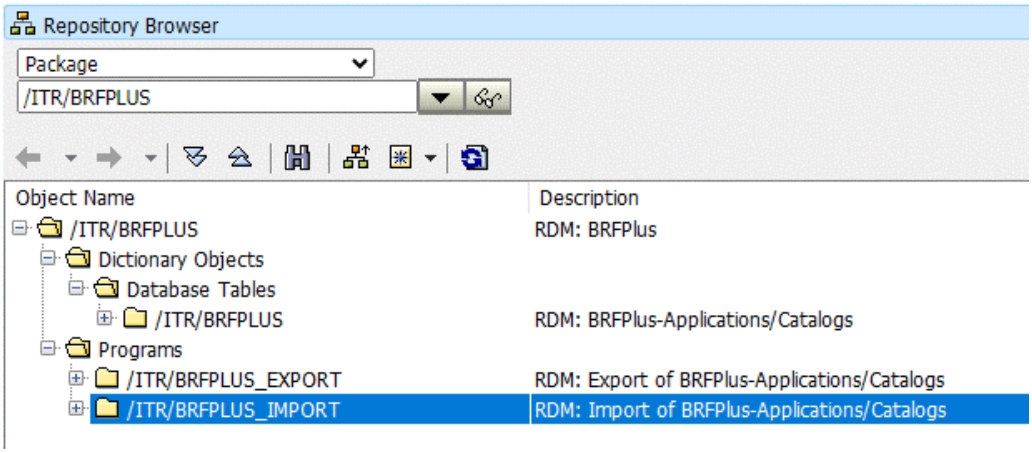
The customizing has to be transferred from client 000 to the operational client where the change requests are processed. SAP standard provides a copy functionality (see also: How to copy BRF+ rules in your target client) and Itego implemented a convenient way to transfer

the customizing using the SAP XML Export/Import functionality of BRF+ (see also: Export and Import of BRFplus XML Data). This is called "Automated Copy" and described below.

3.5.2 Automated Copy

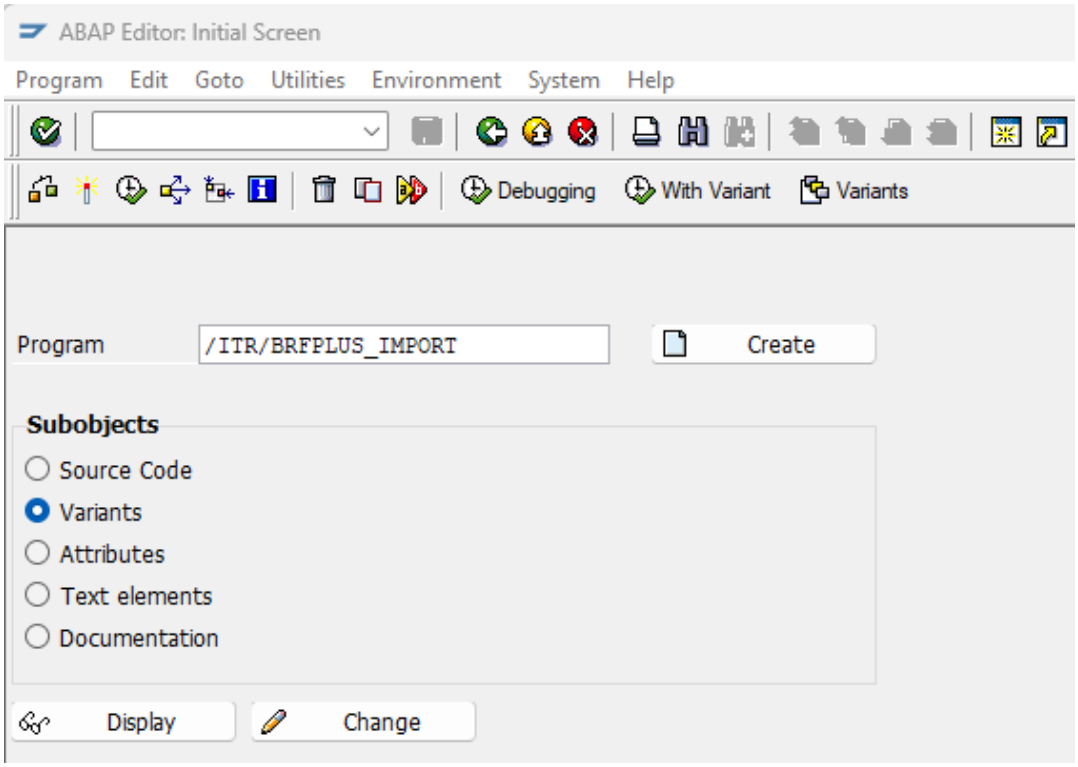
Itego RDM package /ITR/BRFPLUS includes two programs (reports)

- /ITR/BRFPLUS_EXPORT
- /ITR/BRFPLUS_IMPORT

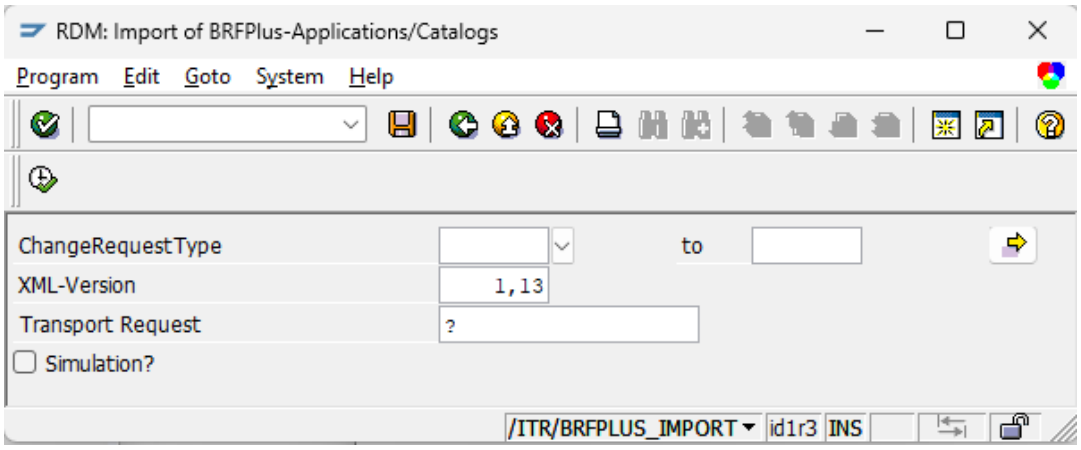


The XML export is already done by Itego and BRF+ data is delivered via table /ITR/BRFPLUS.

Before the XML Import (report /ITR/BRFPLUS_IMPORT) can be started a workbench request has to be created with transaction SE09 or SE10. After creating the request run transaction SA38 and start report /ITR/BRFPLUS_IMPORT:

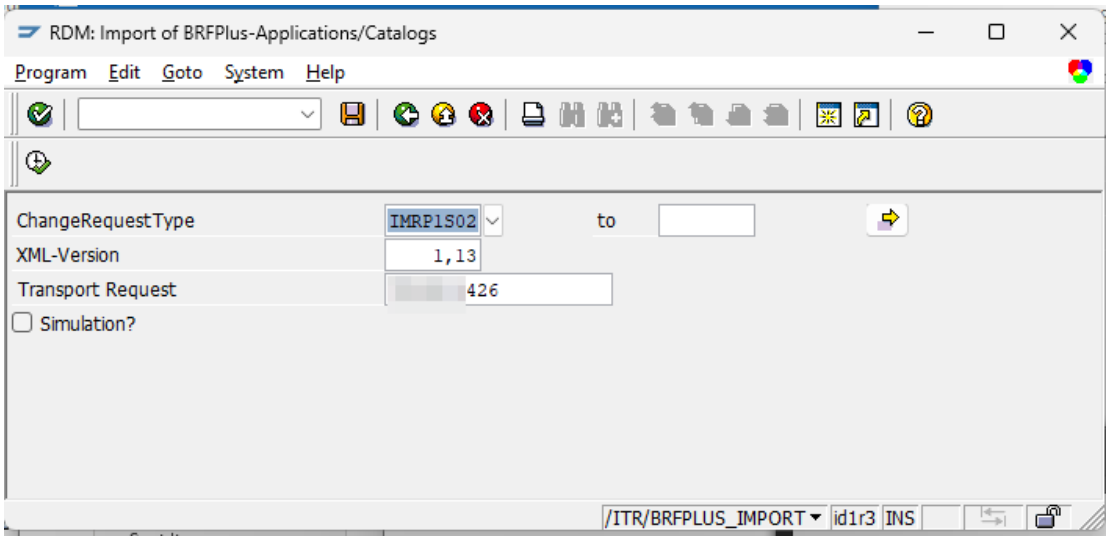

ABAP Editor: Initial Screen
Program Edit Goto Utilities Environment System Help
Debugging
With Variant
Variants
Program
/ITR/BRFPLUS_IMPORT
Create
Subobjects
Source Code
Variants
Attributes
Text elements
Documentation
Display
Change

The report offers the following selection criteria:

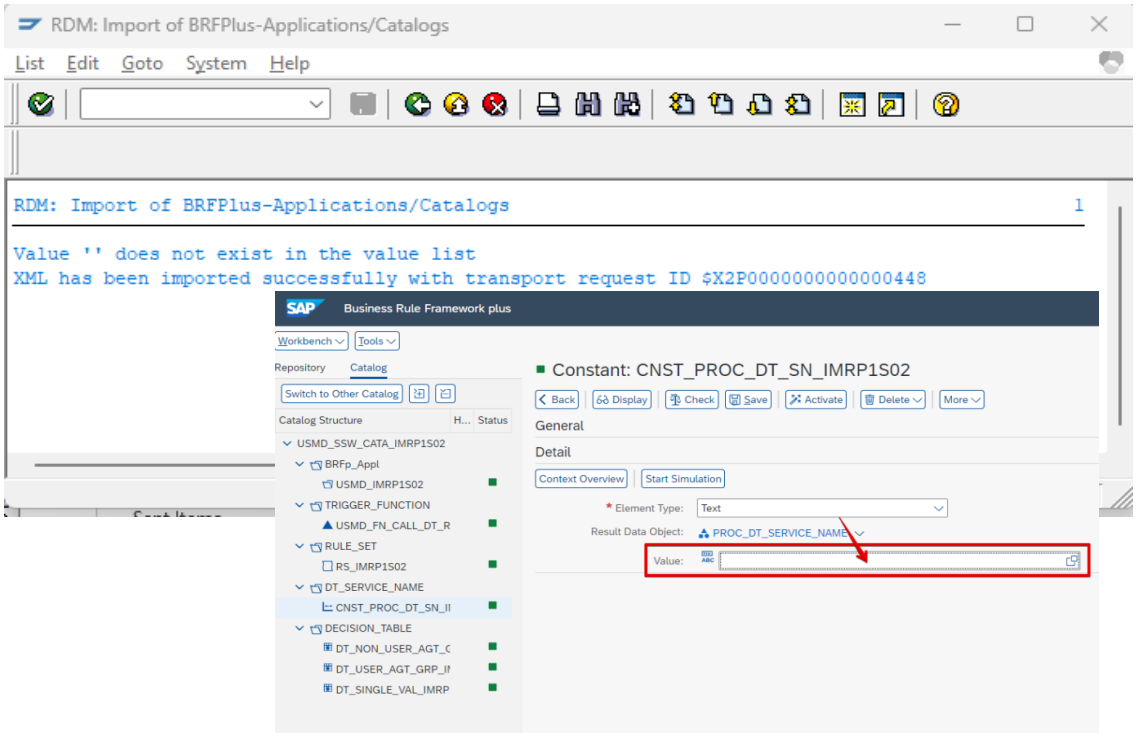


- Change Request Type
  Here you can select a range, use a generic entry like IC* or select explicit change request types to be written to BRF+.
- XML-Version
  Usually it is fine to take the default value.
  - Itego delivers version 1,11; 1,12 and 1,13.
  - A higher version e.g. 1,14 works with 1,13 as well
- Transport Request.
  During the program run, the imported BRF+ data records are written to the transport request (TR) you just created before. The TR is mandatory, as the SAP XML interface requires it. For RDM a TR is not mandatory, but you can use this TR to import the BRF+ data to other systems if necessary.
- Simulation
  Checking this check box means to execute a test run including logging.

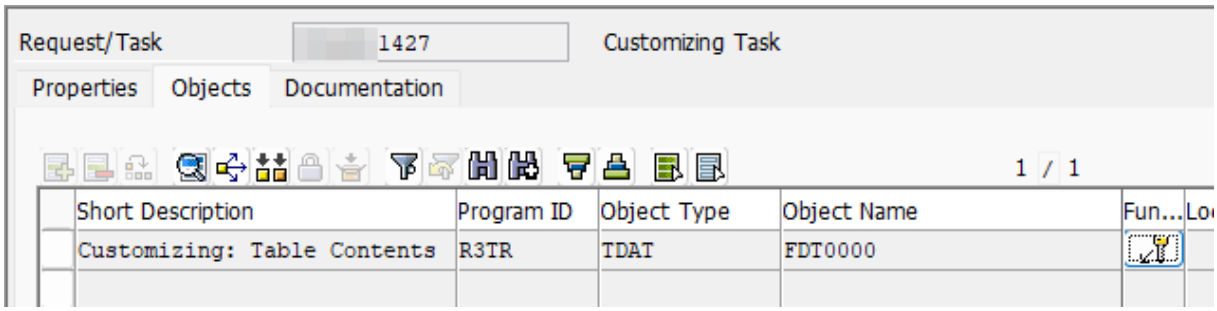
Example: Change request type IMRP1S02



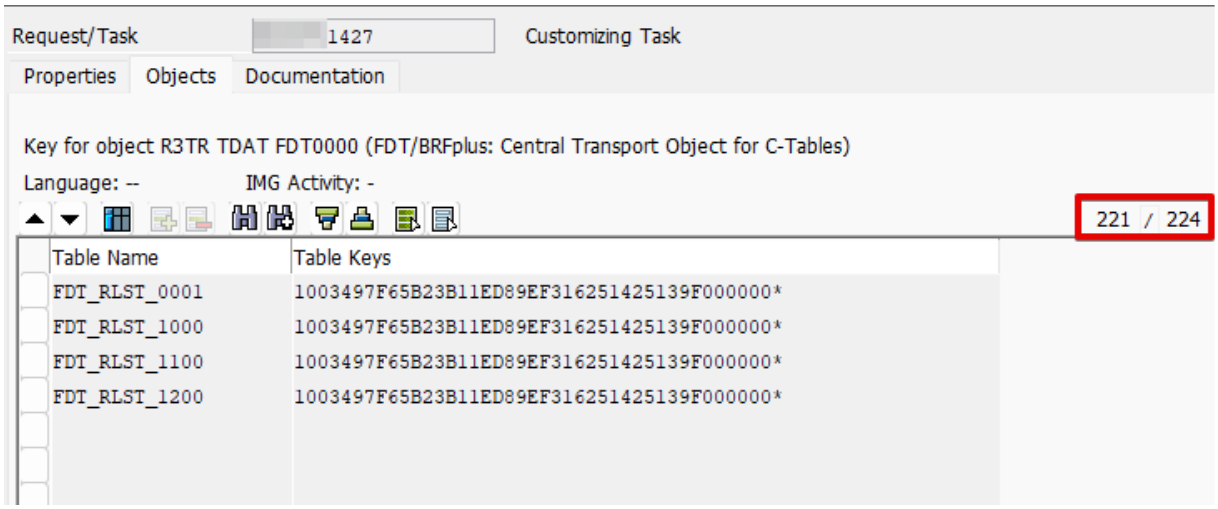
Please ignore the warning:
Value '&1' does not exist in the value list (FDT_EXPRESSIONS-016)
In context of change request decision tables a not needed text field has no value, that's it – please see below:



Your created customizing request has one entry for TDAT FDT0000



and 224 entries for table keys in different tables:



This means a large amount of data will be written to different database tables with a long processing time especially when you copy many (or all) change request types. The XML processing also creates a lot of resource consumption and therefore processing in background is recommended.

3.5.3 Manual Copy

1. **Create Transport**: Log in to client 000 and create a customizing transport.
2. **Identify Application IDs**: In the Data Browser (transaction SE16), enter the table name "FDT_ADMN_0000". In the table-selection screen, search for the name(s) "DT_SINGLE_VAL_I*" and press execute. Then copy all APPLICATION_IDs and close the transaction. Please set the maximum of hits to 1,000.
   Hint: you can use System->List->Save->Save and save the table as a "Text with Tabs" and import this file into a spreadsheet format (like e.g. Excel). Then copy all APPLICATION_IDs.

3. **Transfer Applications to Transport**: Execute transaction SE38 and execute the report "FDT_TRANS". In the Workbench/Customizing field enter the transport, you created in step 1. For the Object ID field, press on multiple selection and paste the copied APPLICATION_IDs from step 3 into the "Single value" table (Hint: you can also save your list of APPLICATION_IDs into a text file and import it by clicking "Import from Text File" in the "multiple selection" dialog box). Then select the "Transport Whole Application(s)" checkbox and execute the transaction. Close the program once it's done.
4. **Release Transport**: Execute transaction SE10 and release the transport, you created in step 1.
5. **Copy Applications to MDG client**: Log on to your MDG client and copy the transport (using transaction SCC1, including the tasks of the request) you just released in step 4 (Source Client: 000). Then check the same table as in step 2 for the copied data.

3.6 Configure Multiple Object Change Request

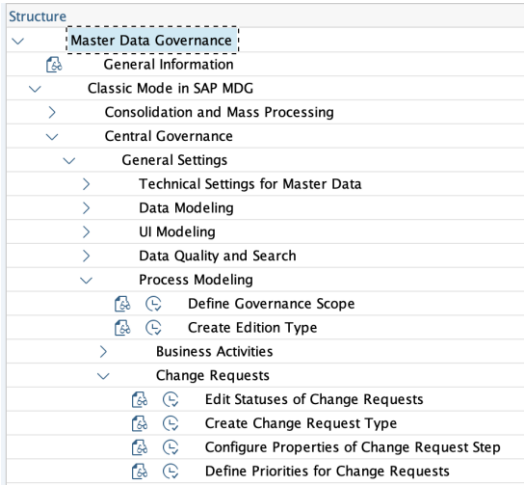
Implementing a multiple object change request allows to manage and process changes for several related master data objects simultaneously within a single workflow transaction. This functionality is recommended for scenarios requiring the coordinated maintenance of interdependent entities (e.g., Company and its Company Codes).

3.6.1 Workflow Settings

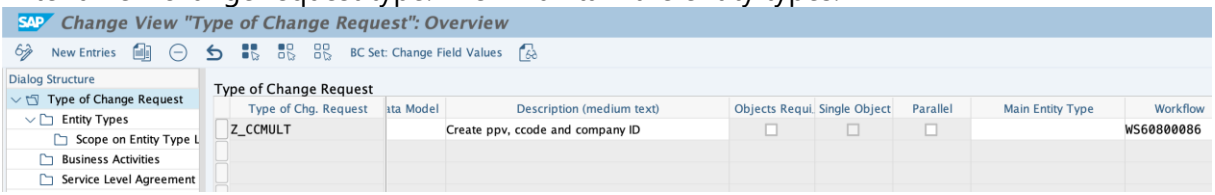
To create a multiple object change request workflow, a new change request type must be created. It is recommended to copy an existing, similar change request type template and modify the settings to ensure consistency and to minimize the configuration effort.

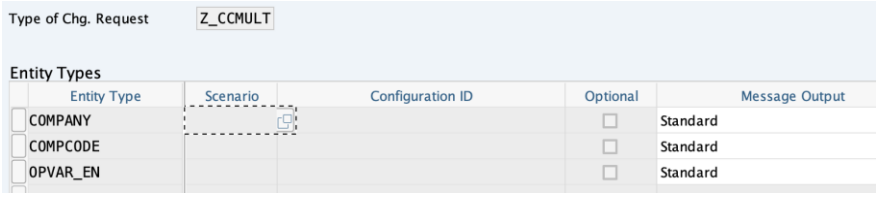
For detailed explanations and more information visit SAP Help Portal – Rule based Workflows
The following short guide shows relevant changes with the example Company Code, Company ID, and Posting Period Variant.

Open Tx MGDIMG and choose "Create Change Request Type".

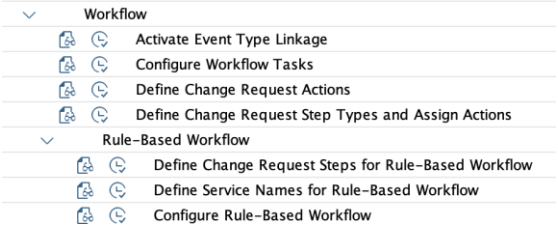


Enter a new change request type. Then maintain the entity types.

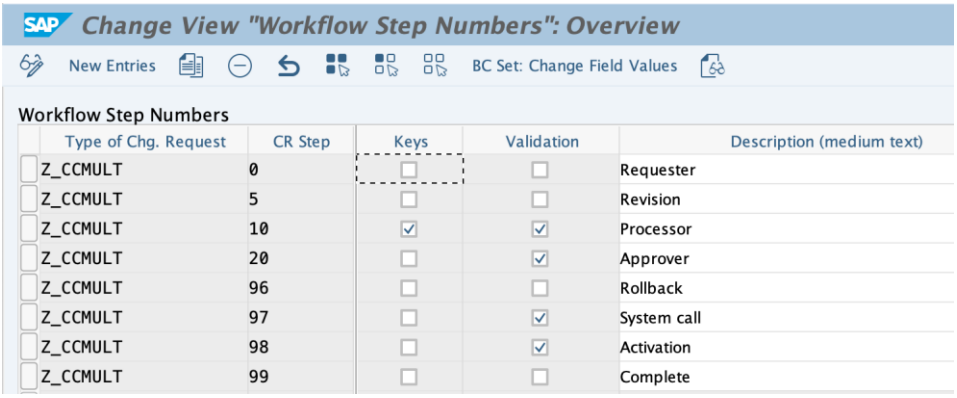




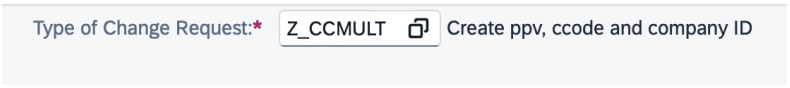
In the overview choose “Define Change Request Steps for Rule-Based Workflow”.



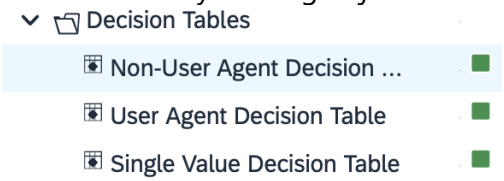
Enter the steps with description for the Change Request Type.



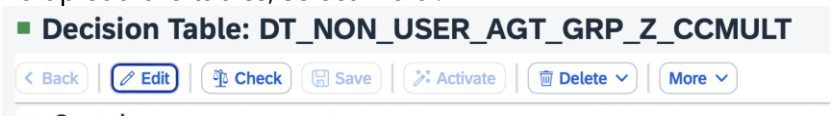
In the overview choose “Configure Rule-Based Workflow”.



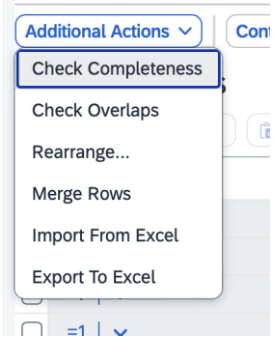
From an already existing object the decision tables should have been downloaded.



To upload the tables, select “Edit”.

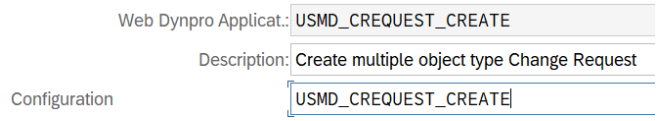


Import the excel for every table save, and activate,



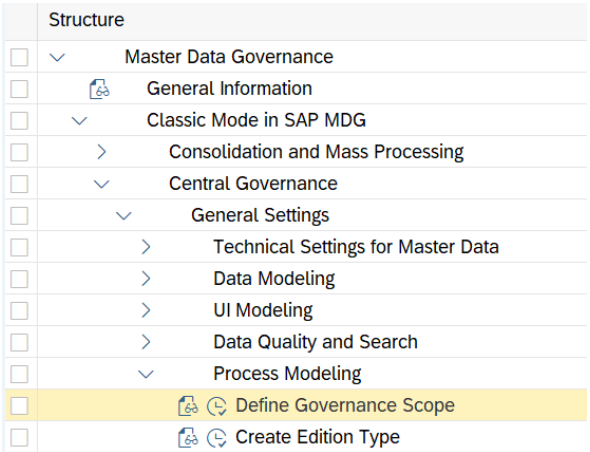
After activation of the workflow can be used via the Fiori tile

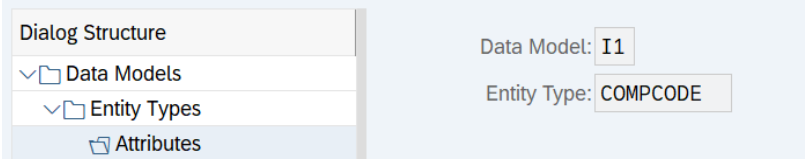
If you are using NWBC the Web Dynpro Application can also be added to a user role or the NWBC launchpad by using these settings:



3.6.2 Governance Settings

In order to make sure that the object types can be assigned to each other in the change request processing check the MDG governance scope and make sure that "yes" is configured. Use transaction MDGIMG:



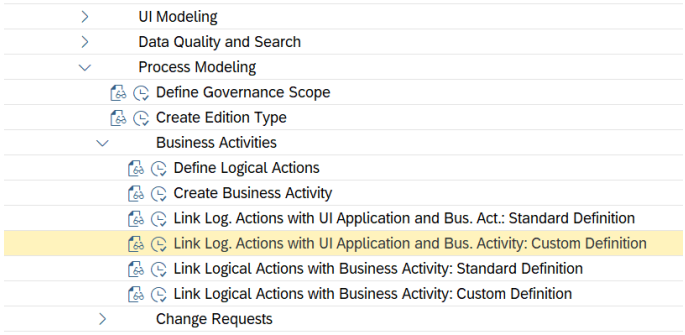


Example here: maintain attributes "Company" and "Posting Period Variant" from "Company Code"

☐	OPVAR	Posting Period Variant	Yes

3.6.3 UI Settings

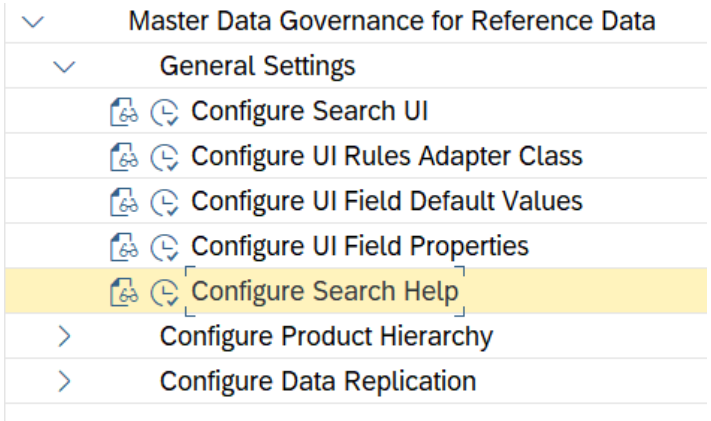
Configuration of User Interface using transaction MDGIMG:



Example: usage of enhanced "Company Code" UI with assignments for "Company" and "Posting Period Variant":

☐	I_COMPANY	*	*	*	USMD_OVP_GEN	/ITR/I1_COMPANY_OVP_GEN
☑	I_COMPCODE	*	*	*	USMD_OVP_GEN	/ITR/I1_COMPCODE_OVP_GEN_02
☐	I_CURRENCY	*	*	*	USMD_OVP_GEN	/ITR/I1_CURRENCY_OVP_GEN

Configuration of searchhelps using transaction: /ITR/RDMIMG



Example here: assign "Company" and "Posting Period Variant" from "Company Code" and assign "Company Code" from "Company":

☑	I1	COMPANY	/ITR/_S_I1_PP_COMPANY	CCODCOMP	Company Codes Address / Company Code	MDG-Staging
☐	I1	COMPCODE	/ITR/_S_I1_PP_COMPCODE	CCODECURR	Company Code / Currency	Customizing
☐	I1	COMPCODE	/ITR/_S_I1_PP_COMPCODE	CCODKTOPL	Company Code / Chart of Accts	Customizing
☐	I1	COMPCODE	/ITR/_S_I1_PP_COMPCODE	CCODLAND1	Company Code / Country/Reg.	Customizing
☐	I1	COMPCODE	/ITR/_S_I1_PP_COMPCODE	CCODSPRAS	Company Code / Language	Customizing
☑	I1	COMPCODE	/ITR/_S_I1_PP_COMPCODE	OPVAR	Company Codes / Posting Period Variants	MDG-Staging
☑	I1	COMPCODE	/ITR/_S_I1_PP_COMPCODE	RCOMP	Company Codes / Global Company Data (for KONS Ledger)	MDG-Staging

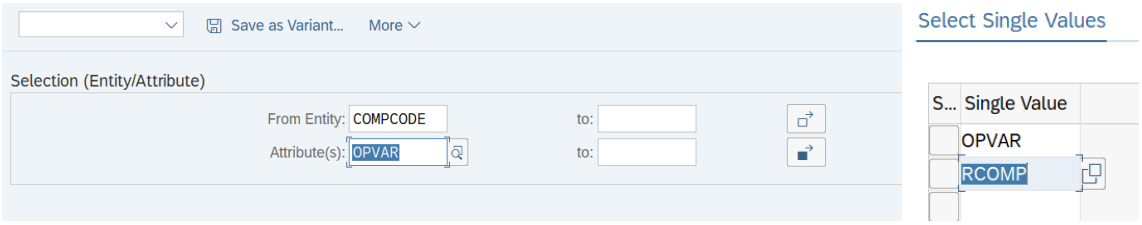
3.6.4 Validation Settings

In some use cases the data model based validations for the so called “existence check” need to be deactivated. For this purpose two reports are delivered which need to be executed after each data model generation

- /ITU/CHG_NO_EXIST_CHECK (for data model attributes)
- /ITU/CHG_NO_EXIST_CHECK_RELA (for data model relations)

The example below is again handling the scenario of maintaining Company, Company Code and Posting Period Variant in a multiple object type change request.

Run report /ITU/CHG_NO_EXIST_CHECK (for data model attributes) with transaction SA38:

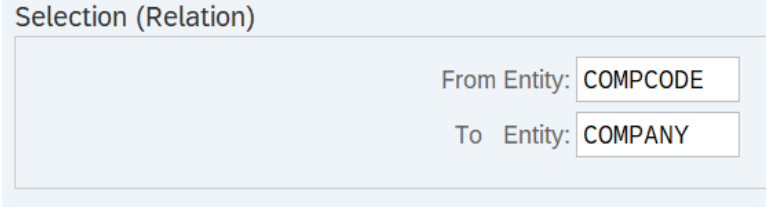


Result:

Entity	Attribute	from	to
COMPCODE	OPVAR	' '	'X'
COMPCODE	RCOMP	' '	'X'

This means: Posting Period Variant (attribute OPVAR) and Company (attribute RCOMP) can now be assigned from Company Code (entity COMPCODE) even though they have not been activated.

Run report /ITU/CHG_NO_EXIST_CHECK_RELA (for data model relations) with transaction SA38:

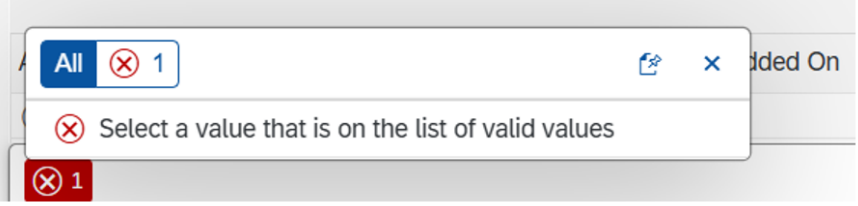


Result:

R_ENTITY	RELATION	TO_ENTITY	OLD_FLAG	NEW_FLAG
COMPCODE	CCODCOMP	COMPANY	' '	'X'

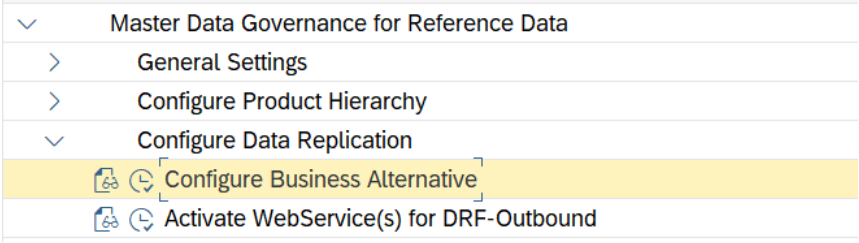
This means: Company Code (entity COMPCODE) can now be assigned from Company (entity COMPANY) can now be assigned from Company Code even though it has not been activated. This is achieved by deactivating the existence check on relation CCODCOMP.

The last two steps need to be executed if the system throws warnings like or similar to this after selecting already



3.6.5 Data Replication and Local Staging Area

In order to be able to select the multiple objects which are maintained in this change request type for activation in the Local Staging Area in the receiving business system it needs to be checked that the change request information is transferred to this system. Check transaction /ITR/RDMIMG in the MDG system:



Example: business system "IH1 client 400":

	Business System	Logical system	Entity Type	A
☐	IH1400	IH1400	RDM_CR_TF	☑

3.7 Adjust Governance Scope

Adjusting the Governance Scope on SAP MDG leads to a reduced set of attributes which are maintained for a specific object type. As the set of attributes might also differ from a local point of view also the local scope on SAP Receivers can be adjusted.

3.7.1.1 Define Governance Scope

You can determine a set of governed attributes for each reference data object type. Fields which will be defined as "out-of-scope" are shown as read-only in change requests, unless they are removed from the user interface.
Prerequisite: You are aware of the consequences of changing the governance scope. See the help document in transaction MDGIMG "Customizing for Master Data Governance": General Settings -> Process Modeling -> Define Governance Scope before you execute this activity.

Most fields defined in this customizing activity will not be replicated but will be replaced by a "no data" sign which will allow to keep the local field values in the receiving system. Certain limitations apply, so please make sure to test the data replication after the definition of the Governance Scope.

3.7.1.2 Configure Search Helps

This feature is mainly used to avoid additional configuration or development efforts when the Search Helps in an object type which is in the RDM scope refer to an object type which is not in the RDM scope.

Note: currently some object types are not fully covered. This in particular includes all object types which in the Search Help refer to the object types language, country and currency (which could be loaded to RDM) but also Unit of Measure, Tax Code, MRP Controller and Holiday Calendar (which might need additional configuration for some Search Helps). Please consider to clarify remaining questions by contacting support@itego.de (subject: "Configure Search Helps")

Depending on the chosen Governance scope you should align the search help for each object type in scope. This can be done by calling transaction /ITR/RDMIMG -> General Settings -> Configure Search Helps. Here the following pattern should be applied:

- Configure Search Helps for each object type in scope
- For each object type configure all search helps
- For each search help configure:
    - "Customizing" if the object type to be maintained in this attribute **is not** part of the Governance Scope

- "MDG-Staging" if the object type to be maintained in this attribute **is** part of the Governance Scope

RDM will show

- Data from the ERP persistence (related ERP customizing tables) for each search help flagged with "Customizing"
- Data from the MDG persistence (generated MDG tables) for each search help flagged with "MDG-Staging"

Note: if you do not configure this RDM will show data as defined in the data model. Means: sometimes data will be shown based on generated MDG tables, sometimes data will be shown based on related ERP customizing tables. As a data replication to the MDG client should be configured (see Chapter 3.10), data in the ERP customizing tables in general should be the same as the data in the generated MDG tables. Which means that the entry "Customizing" in most cases should be the right setting because:

- for the Business Governance Alternative (Direct Update) the data is written to the ERP persistence immediately after activating an approve Change Request
- for the IT Governance Alternative (via Local Staging Area) the data is written to the ERP persistency immediately after the data is activated in the Local Staging Area.

However, if you configure change request types which can be used to maintain more than one object type, the expected results could be different as the search help in such a change request type usually would show inactive data as well. In these cases, the setting "MDG-Staging" might make sense.

3.7.2 SAP Business System: Adjust Local Scope

Based on the attributes which can be maintained globally on the SAP MDG Sender you can define specific scope adjustments on each SAP Receiver system. For this use transaction /ITR/RDM_LSA_FIELDS which maintains view /ITR/LSA_TABS_FV. Every attribute which is not within this scope definition will not be overwritten when data is activated on the business system.

3.8 Adjust User Interfaces

User Interfaces should be adjusted based on the Governance Scope and the user requirements.

The Master Data Framework (MDF) provides capabilities for the definition of

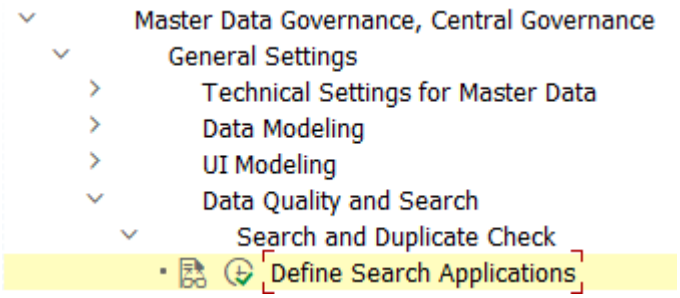
- Field properties

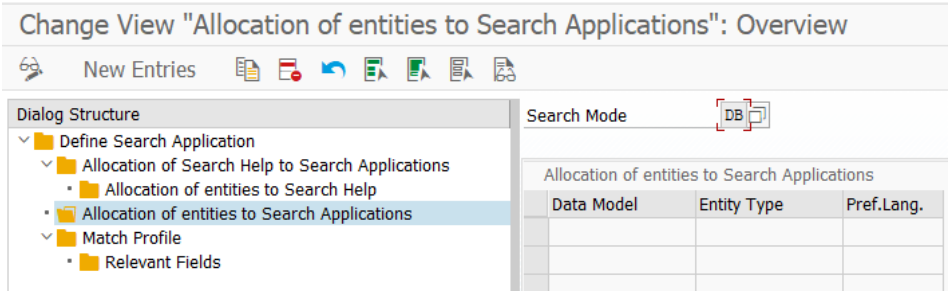
- Search configurations
- Default Values
- Validations and
- Derivations

Please check the document "Technical Documentation of the MDF Configuration Management" to learn more about this.

On top of these functionalities provided, the SAP Floorplan Manager (FPM) can be leveraged for additional adjustments.

Note: In order to define Search Modes for individual entities SAP provides the configuration option below. Make sure that you are aware of side effects to other entities described below.





Make sure that this configuration only is used when you define an allocation for every entity type used in and SAP MDG application. Otherwise, an entry for an RDM object might impact the search for other entities. For example, the search mode for a SAP MDG-F based object like Cost Center, which should be HA (HANA) based, might be changed to DB (Database) by the SAP MDG framework.

3.9 Configure Rule Based Workflow

Prerequisite: You have configured the general settings for SAP Business Workflow in Customizing for SAP NetWeaver under Application Server -> Business Management -> SAP Business Workflow.

The configuration of the rule-based workflow is done in transaction MDGIMG: General Settings -> Process Modeling -> Workflow -> Rule-based Workflow -> Configure Rule-Based Workflow.

Consider the following BC Sets from section 3.3 "Activate Business Configuration Sets" as templates for your configuration (**check also the notes mentioned in this section**):

- /ITR/MDG_RDM_CREQUEST_<n> MDG RDM Change Request Types
- /ITR/MDG_RDM_WORKFLOW_<n> MDG RDM Workflow

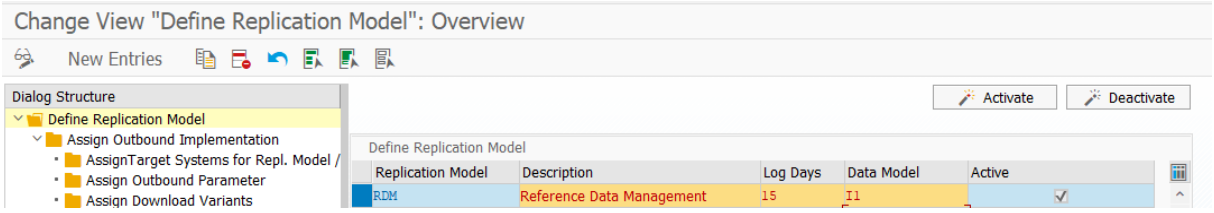
3.10 Set Up Data Replication

In order to configure the Data Replication for RDM the "Data Replication Framework" (DRF) needs to be configured as well as the technical settings for ALE or Webservices.

Note: Make sure that you also define the data replication to the MDG client which will make sure that related ERP customizing tables are synchronized. For the IT Governance Alternative this should be configured via the Local Staging Area in the MDG development system whereas for the Business Governance Alternative this should be configured by using the direct replication to the MDG production system (with optional connection to the test as well as the development system). See chapter 5.1.5 for a configuration example.

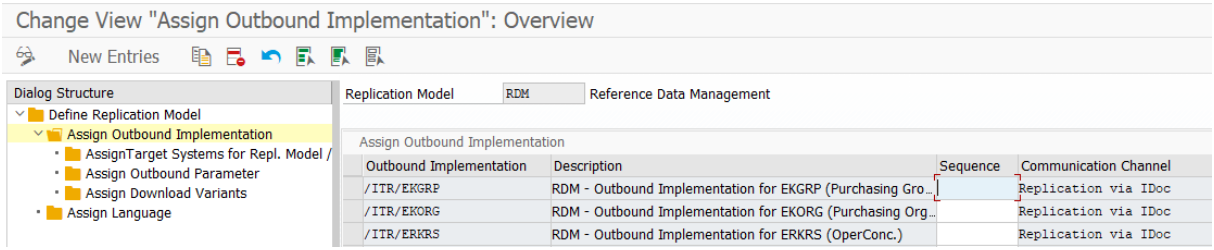
3.10.1 Data Replication Framework settings

Use transaction DRFIMG to maintain Business Systems and the Data Replication Model.



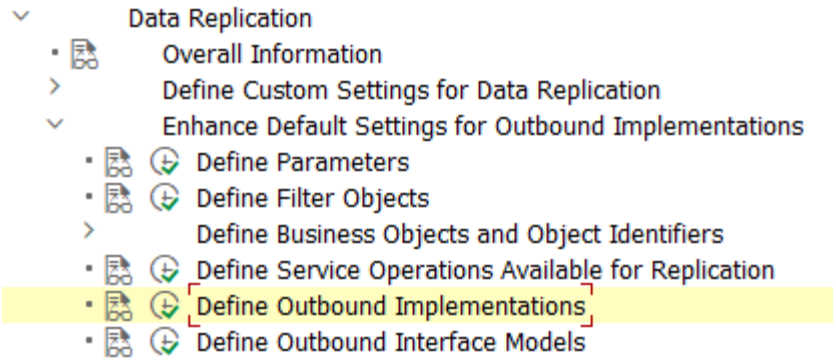
3.10.1.1 Configuration of Outbound Implementations

Configure Outbound Implementations for your model based on the scope of your RDM implementation and define a communication channel.



For some RDM object types the assigned outbound implementations follow an object specific implementation. If you do not have project specific requirements which require the usage of these, please change the implementation classes for the following objects to the generic implementation:

Transaction: DRFIMG



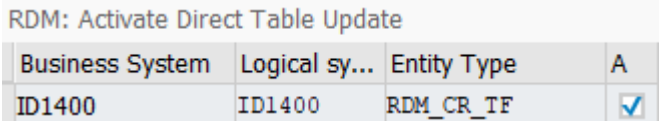
Object Type	**Outbound Implementation**	**Outbound Implementation Class**
Currency	/ITR/CURRC	/ITR/CL_CURRENCY_OUT_IDOC_GEN
Purchasing Group	/ITR/EKGRP	/ITR/CL_EKGRP_OUT_IDOC_GEN
Purchasing Organization	/ITR/EKORG	/ITR/CL_EKORG_OUT_IDOC_GEN
Account Group Customer	/ITR/KTOKD	/ITR/CL_KTOKD_OUT_IDOC_GEN
Account Group Vendor	/ITR/KTOKK	/ITR/CL_KTOKK_OUT_IDOC_GEN
Account Group Gen. Led.	/ITR/KTOKS	/ITR/CL_KTOKS_OUT_IDOC_GEN
Country	/ITR/LAND1	/ITR/CL_LAND1_OUT_IDOC_GEN
Material Group	/ITR/MATKL	/ITR/CL_MATKL_OUT_IDOC_GEN
Unit of Measure	/ITR/MSSIE	/ITR/CL_MSSIE_OUT_IDOC_GEN
Payment Term Day Limit	/ITR/PAYMD	/ITR/CL_DAYLIMIT_OUT_IDOC_GEN
Payment Term	/ITR/PAYMN	/ITR/CL_PAYMNTTRM_OUT_IDOC_GEN
Plant	/ITR/PLANT	/ITR/CL_PLANT_OUT_IDOC_GEN
Product Hierarchy	/ITR/PRODH	/ITR/CL_PRODH_OUT_IDOC_GEN

Product Hierarchy	/ITR/PRODL	/ITR/CL_PRODHL_OUT_IDOC_GEN
Product Hierarchy	/ITR/PRODN	/ITR/CL_PRODHN_OUT_IDOC_GEN
Region	/ITR/REGIO	/ITR/CL_REGIO_OUT_IDOC_GEN
Sales Org	/ITR/SAORG	/ITR/CL_SALESORG_OUT_IDOC_GEN

Note: You do not have to change the outbound implementation classes for other objects. They follow a generic approach even if the classes do not have a "_GEN" suffix.

3.10.1.2 Replication of change request related information

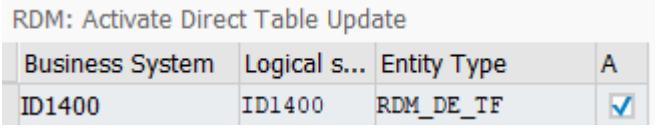
Change Request related information will be replicated to target systems by configuring the parameter RDM_CR_TF using /ITR/RDMIMG -> Configure Data Replication -> Configure Business Alternative. This configuration is target system dependent and needs to be configured for each target system if required for all systems. Example for a target system called "ID1400":


RDM: Activate Direct Table Update

Business System	Logical sy...	Entity Type	A
ID1400	ID1400	RDM_CR_TF	☑

3.10.1.3 Deletion of assignments

Deletion of assignments will be replicated to target systems by configuring the parameter RDM_DE_TF using /ITR/RDMIMG -> Configure Data Replication -> Configure Business Alternative. This configuration is target system dependent and needs to be configured for each target system if required for all systems. This for example allows to delete language dependent texts or an entry in the sequence of the object type "Planning Scope". Example:


RDM: Activate Direct Table Update

Business System	Logical s...	Entity Type	A
ID1400	ID1400	RDM_DE_TF	☑

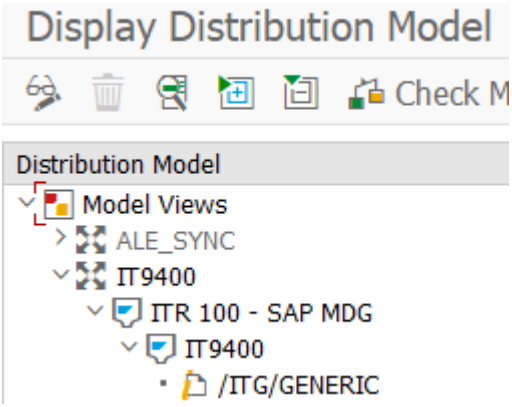
Note: Please be aware that these deletions are not supported for all object types in all cases. Please consider to clarify remaining questions by contacting support@itego.de (subject: "Deletion of assignments").

3.10.2 Data Replication using Application Link Enabling (ALE)

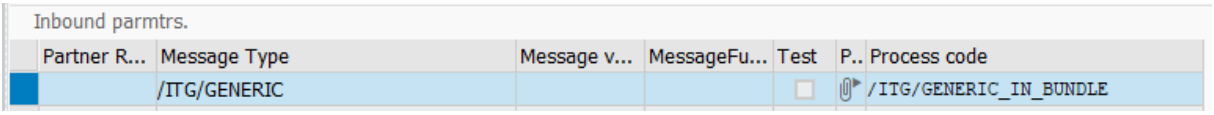
If the Data Replication for reference data objects shall be implemented using ALE the communication needs to be setup between the RDM system and the receiving systems.

3.10.2.1 ALE Configuration for Generic Message Type

For most object types RDM uses the generic message type /ITG/GENERIC which needs to be configured using transactions BD64 and WE20. The figures below show the most important settings. For a more detailed description please have a look at the configuration example provided in chapter 5.1 Data Replication based on SAP ALE.
BD64 (example)



WE20 (example from receiving system)



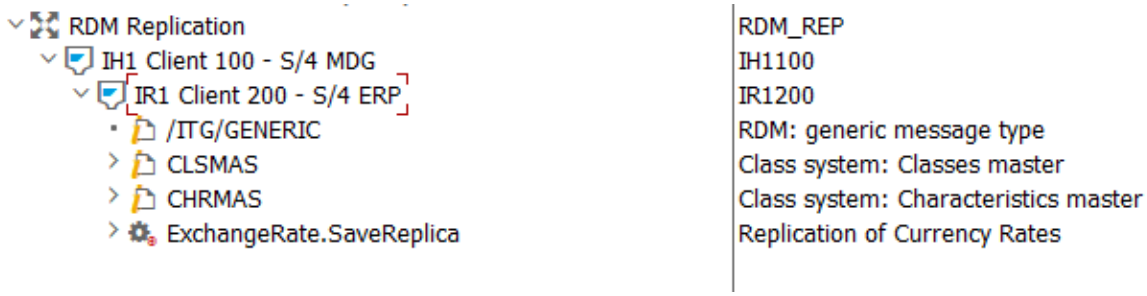
3.10.2.2 ALE Configuration for Object Specific Message Types

For some RDM object types SAP delivers standard message types which need to be configured. These are:

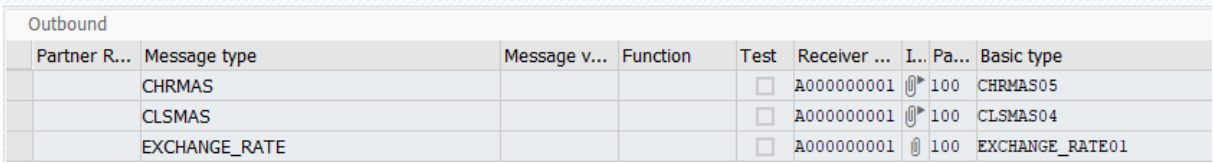
- Classification (Classes and Characteristics): CLSMAS and CHRMAS
- Exchange Rates: EXCHANGE_RATE

If these objects are part of you implementation scope, please add these message types as shown in the previous chapter.

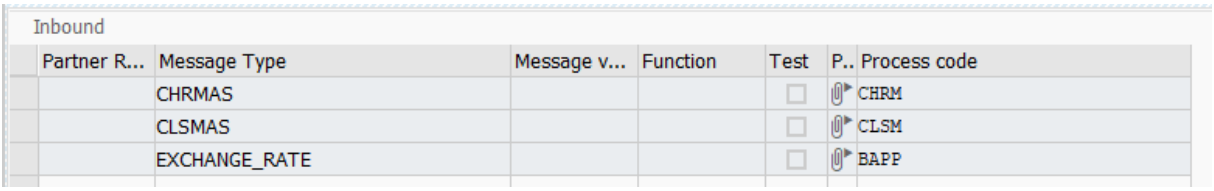
Example from Transaction BD64 (Sender):



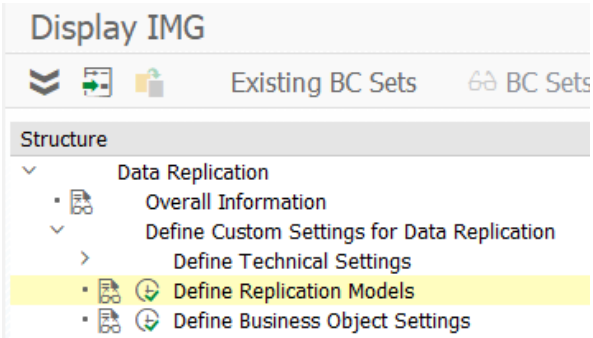
Transaction WE20 (Sender), after generation of partner profiles:

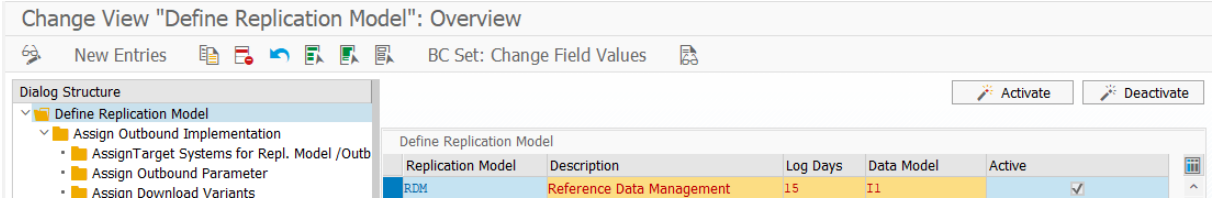


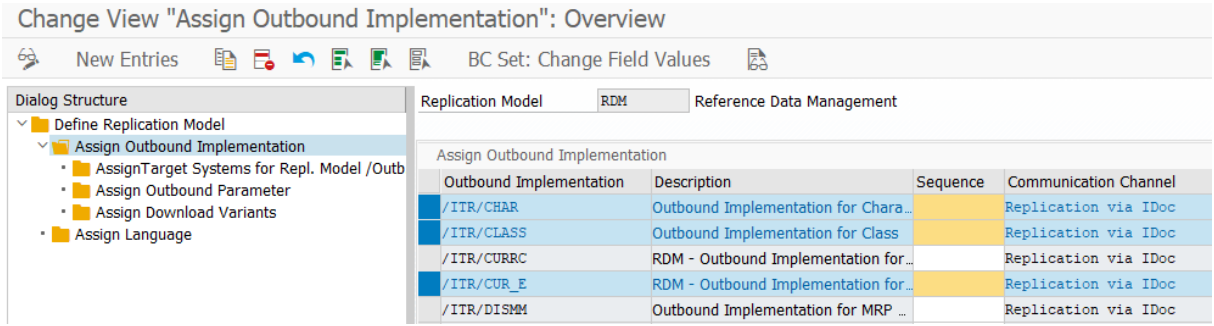
Transaction WE20 (Receiver), after generation of partner profiles:



For these please also make sure that they have been added in the Replication Model (transaction DRFIMG):







3.10.3 Data Replication using WebServices

RDM implements SOAP as a network protocol, which uses XML to transfer the reference data. With this it enables an exchange of data between heterogeneous applications on different systems. The web services are described in WSDL files (Web Service Description Language) which are provided by the server. Using this information, the client application obtains information about the offered web services.

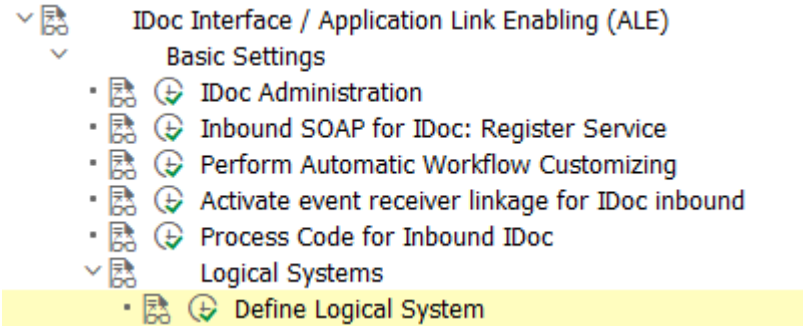
The web services offer two functionalities:

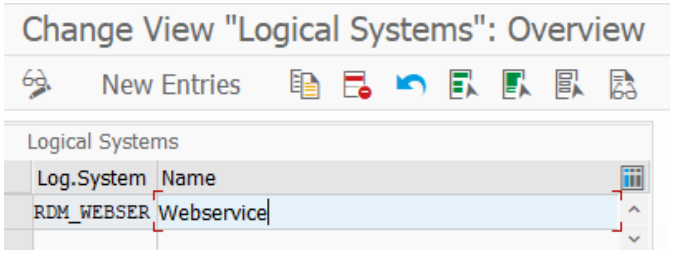
- Push (Send from RDM)
- Pull (Call from outside)

Examples:

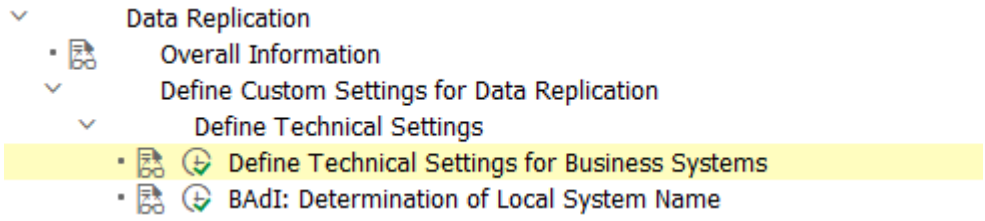
- Push (generic): /ITR/RDM_WS_SEND_OBJECT
- Pull (generic): /ITR/RDM_WS_GET_OBJECT
- Pull (object-specific): /ITR/RDM_WS_GET_PLANT

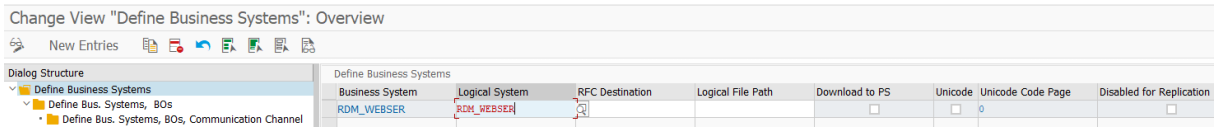
Transaction SALE: Define Logical System

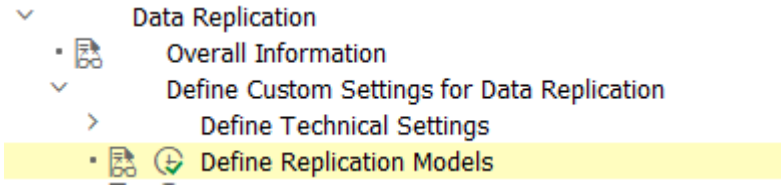


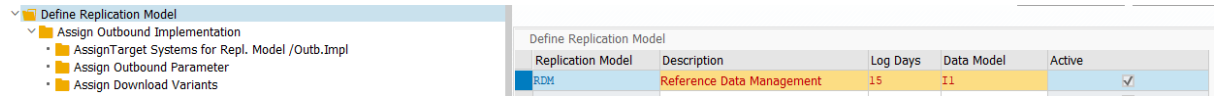


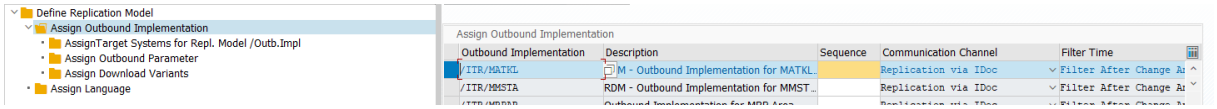
Transaction DRFIMG: Define Technical Settings and Replication Models

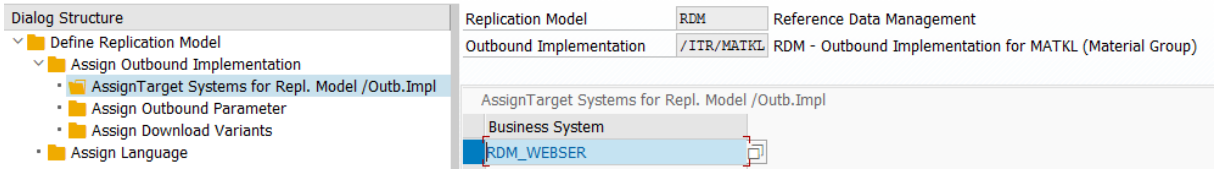


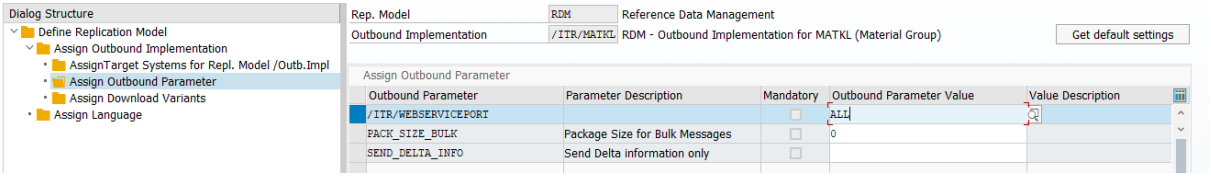






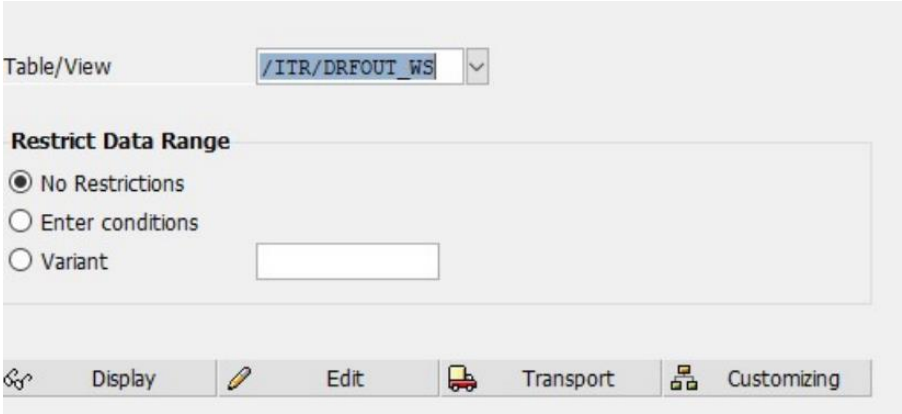


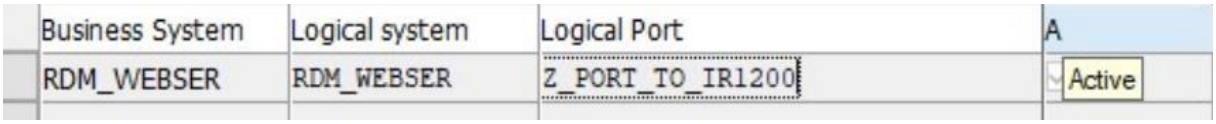




Using outbound parameter /ITR/WEBSERVICEPORT you can define which logical port will be used. If you define Value "ALL" all ports defined will be used.

Maintenance of ports via table /ITR/DRFOUT_WS





For necessary settings with regards to the WebService Environment using transaction SOAMANAGER please have a look at chapter 5.2 Data Replication based on Webservices.

For more information, please contact support@itego.de (subject: "Configuration Guide RDM: Webservices")

3.10.4 Key Mapping

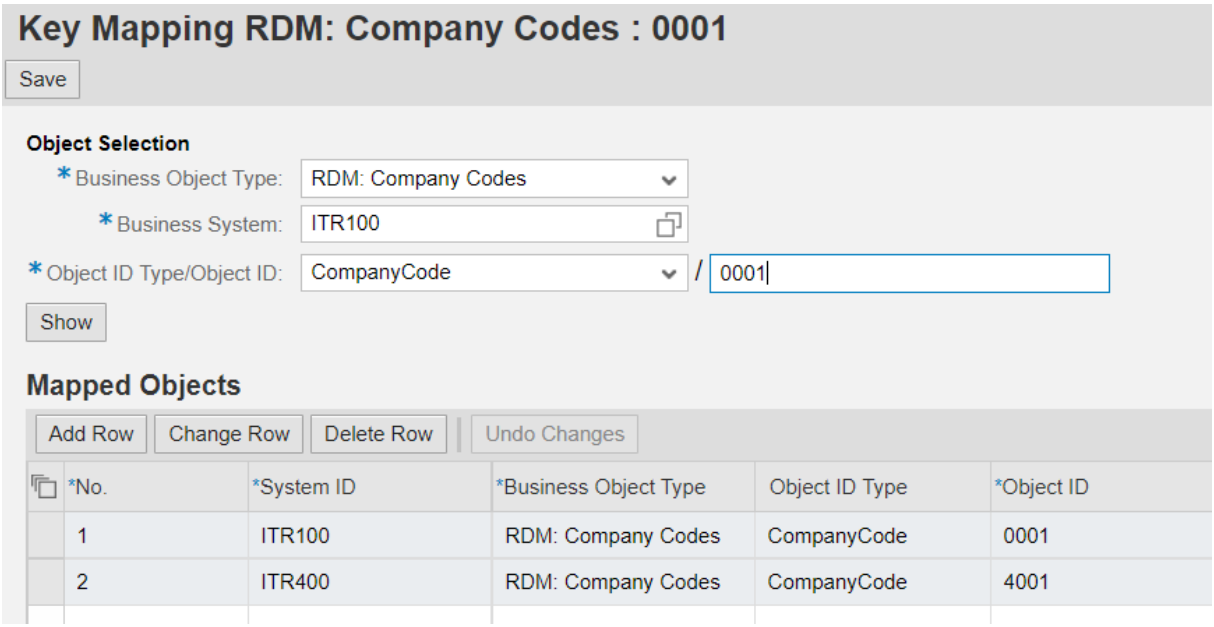
SAP MDG provides the following transactions to maintain and analyze Key Mapping:

- MDG_KM_MAINTAIN - Maintain Key Mapping
- MDG_ANALYSE_IDM - Search Key Mapping

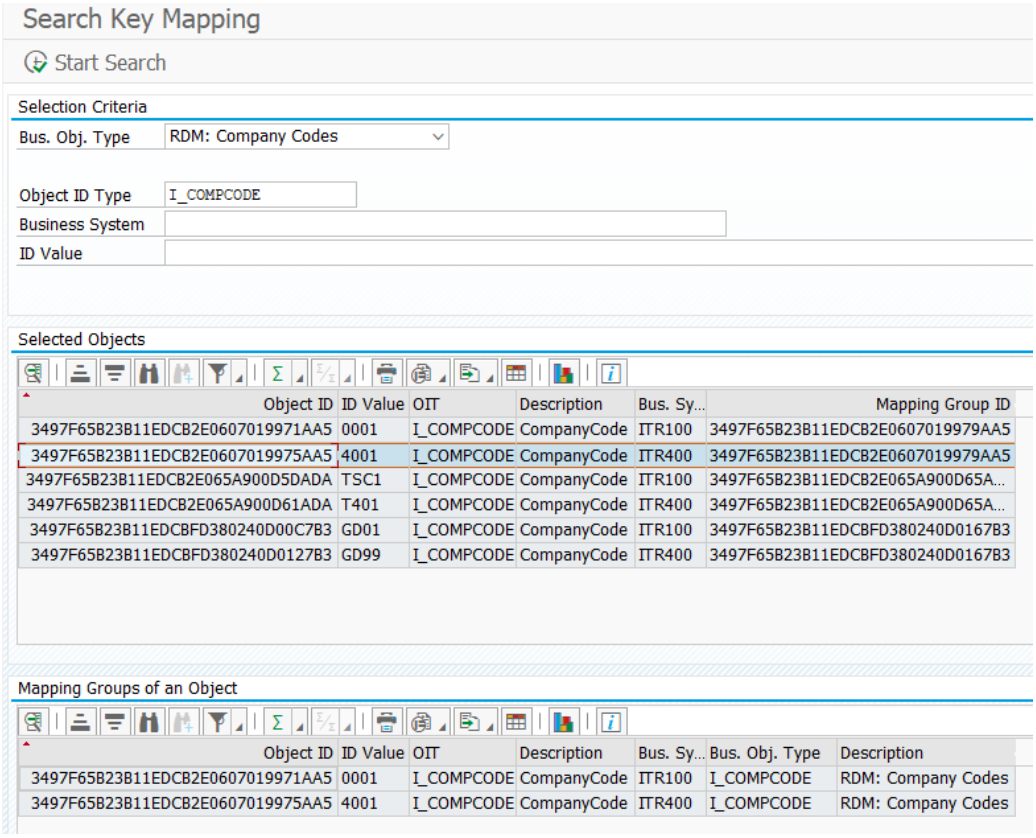
Use these transactions to maintain or search the Key Mapping for reference data object types by using:

- Business Object Type: "RDM: <Object Type>". Example: "RDM: Company Codes"
- Object ID Type: "I_<Entity Type> Description". Example: "I_COMPCODE CompanyCode"

Example for maintenance of Key Mapping:



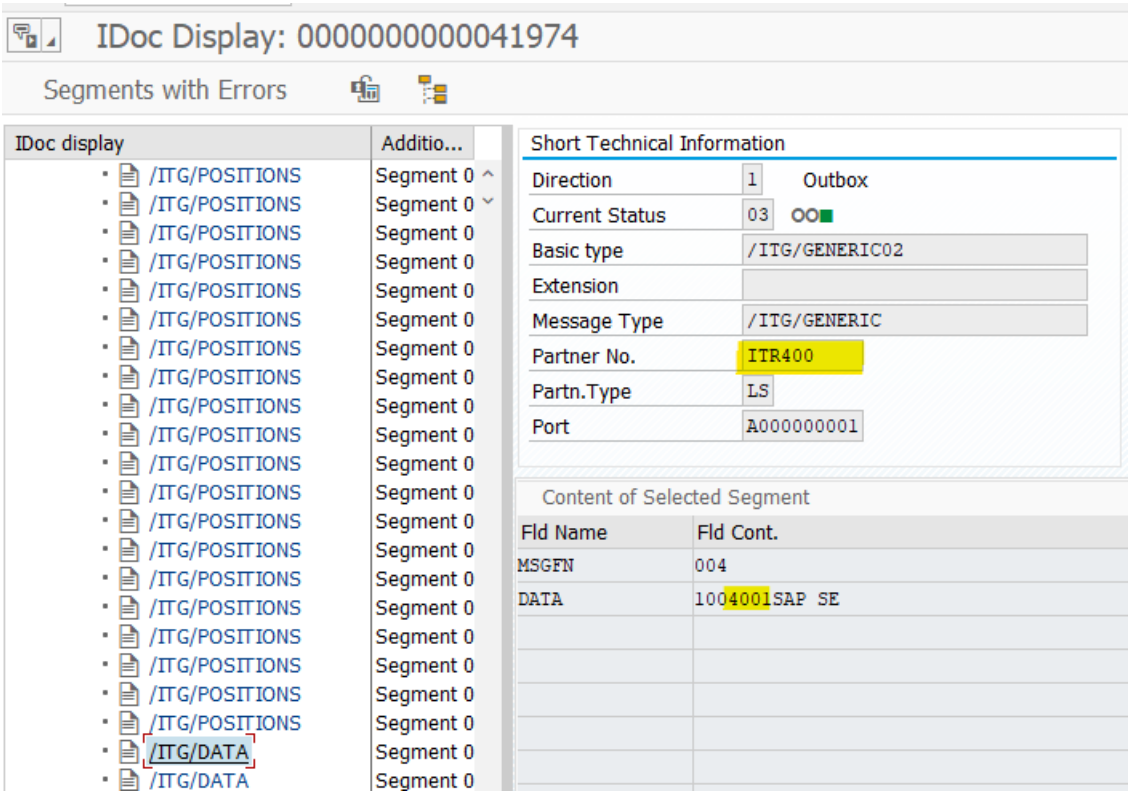
Example for Search Key Mapping:



Export to Spreadsheet:

	B	E	G
1	ID Value	Business System	Description of Business Object Type
2	0001	ITR100	RDM: Company Codes
3	4001	ITR400	RDM: Company Codes

In the data replication the maintained values will be mapped based on the receiver system. Example, based on the maintained mappings above (transaction BD87):



3.10.5 Value Mapping

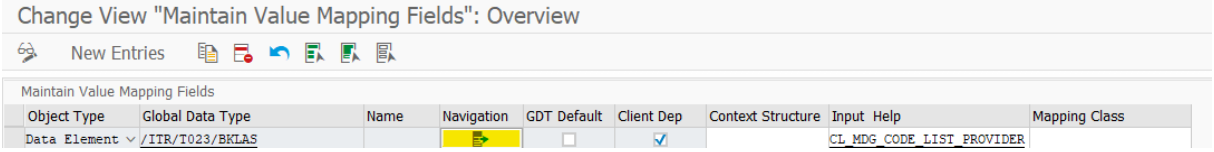
SAP provides the transaction "VMIMG - Value Mapping Customizing" to maintain Value Mappings. Use this transaction to maintain Value Mapping for reference data object types by using:

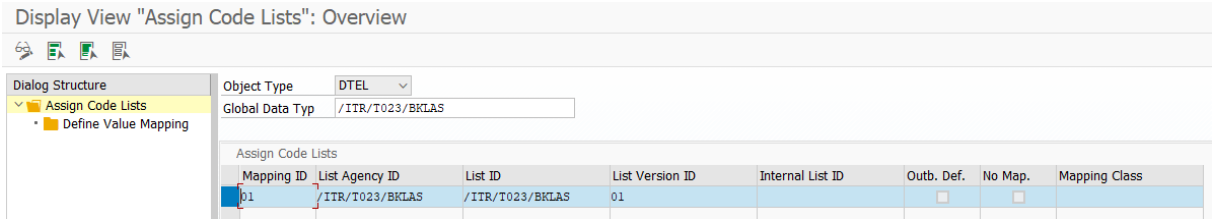
- Type: Data Element
- Global Data Type (GDT): "/ITR/<table>/<attribute>". Example: "/ITR/T023/BKLAS" (Valuation Class for Material Group).

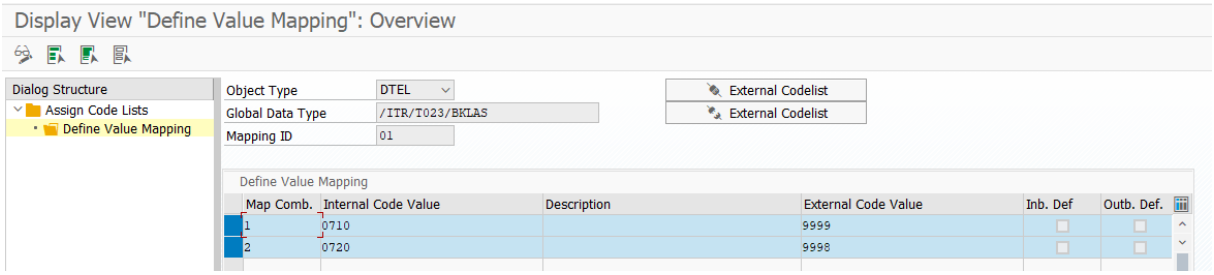
Examples for other attributes:

- List ID: "/ITR/T023/BKLAS"
- List Agency ID: "/ITR/T023/BKLAS"
- List Version ID: 01

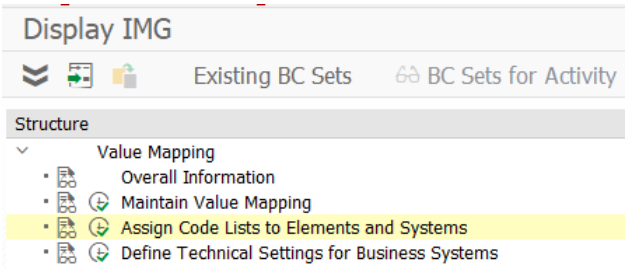
Maintained example:

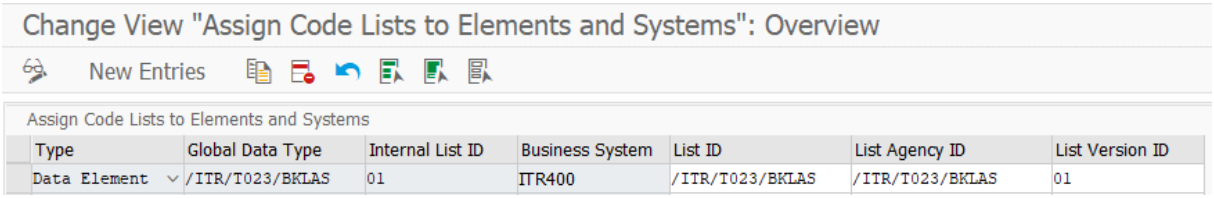






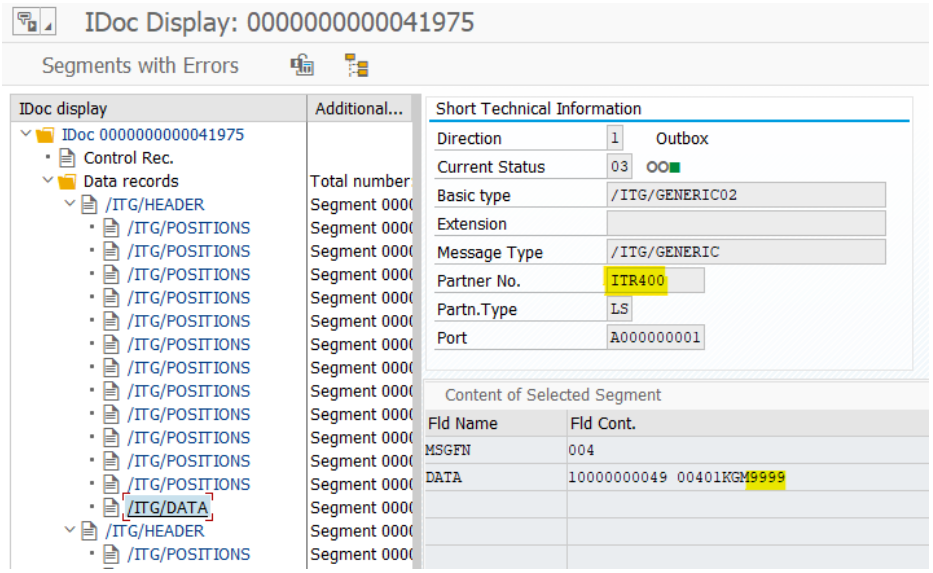
External codes (for system ITR400) for internal codes. Example: 9999 for 0710.



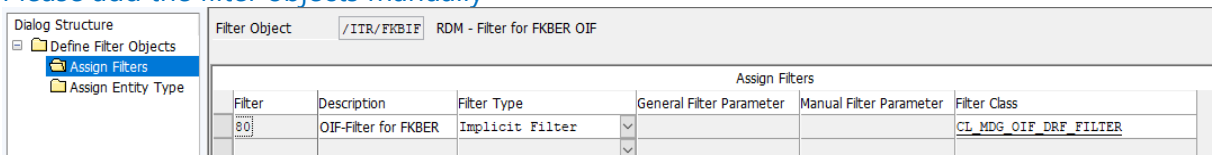


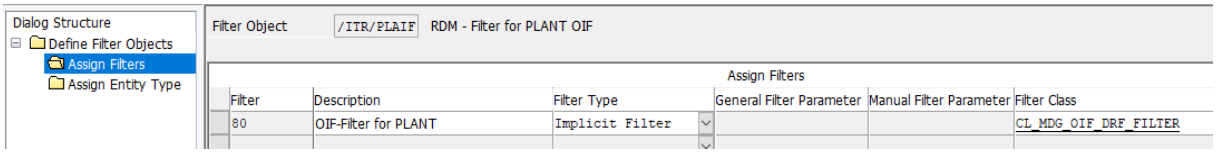
Code list are now in this example assigned for Business System ITR400. Make sure that this entry is added after you added the mapping in step “Maintain Value Mapping”. Otherwise the system will not allow you to add this entry here.

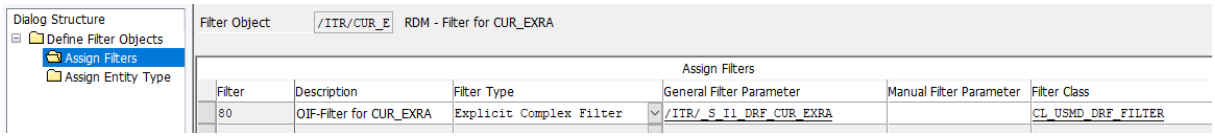
In the data replication the maintained values will be mapped based on the receiver system. Example, based on the maintained mappings above (transaction BD87):



Please add the filter objects manually







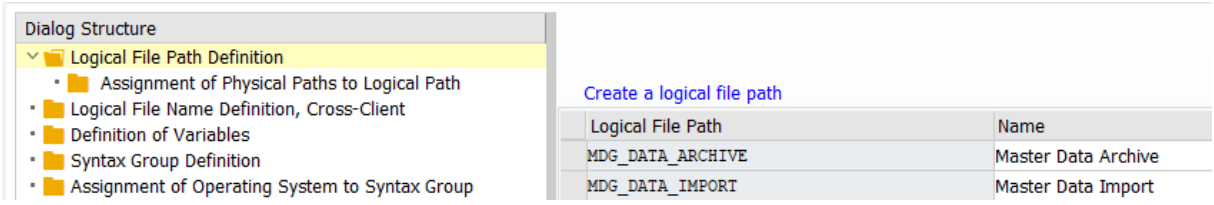
3.11 Set Up Data Transfer

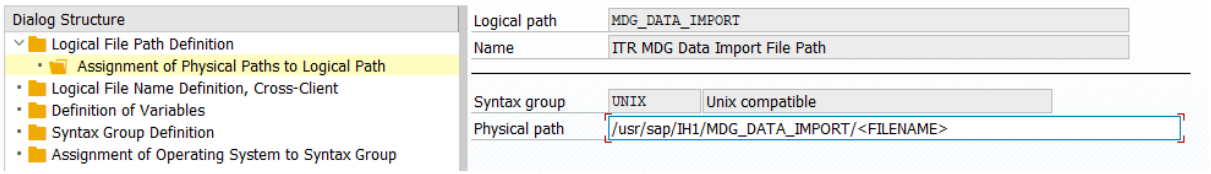
Data Transfer needs to be configured for initial load and consist of the configuration of MDMGX (on an SAP business system) and Data Import (on the MDG RDM system).

The configuration of MDMGX is done through the activation of the BC Sets /ITR/RDM_MDMGX_<n> (see: 3.3 "Activate Business Configuration Sets") on the SAP business system. This enables users to extract reference data objects from this system using transaction MDMGX.

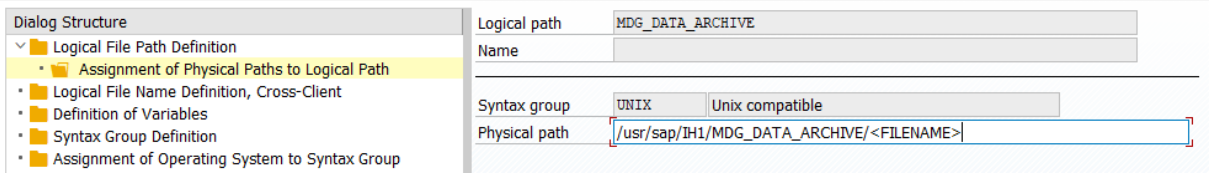
Please also check section 3.3 "Activate Business Configuration Sets" for the activation of the Data Transfer on the MDG system (BC Sets /ITR/MDG_RDM_DT_<n>). These BC Set deliver the necessary object type definitions. After this configure Data Transfer in transaction MDGIMG: Data Transfer -> Define File Source and Archive Directories for Data Transfer. For details check the IMG documentation for this activity and the additonal configuration example below.

Transaction FILE:

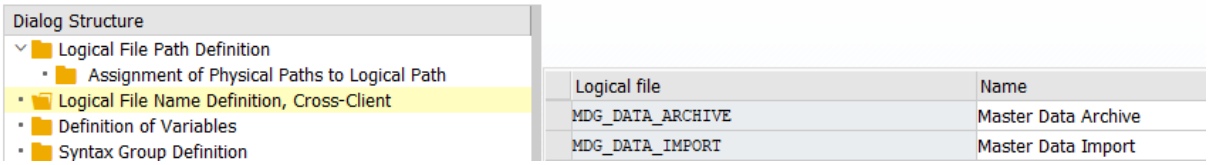


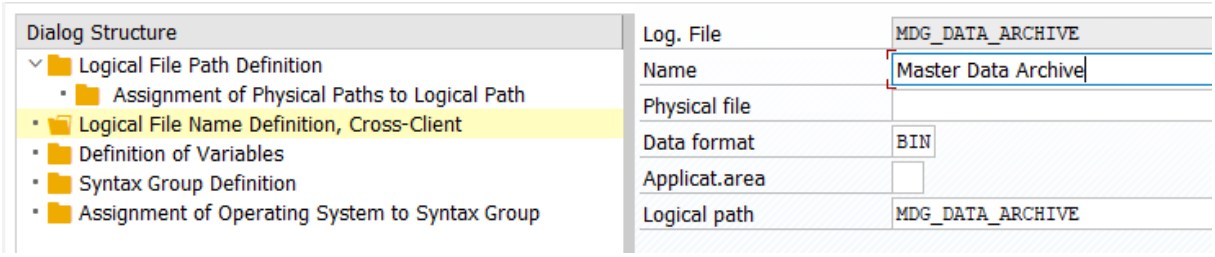


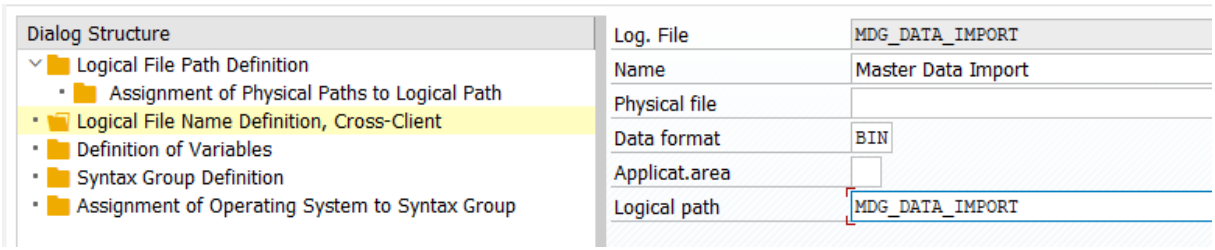
e.g.: /usr/sap/<systemID>/MDG_DATA_IMPORT/<FILENAME>



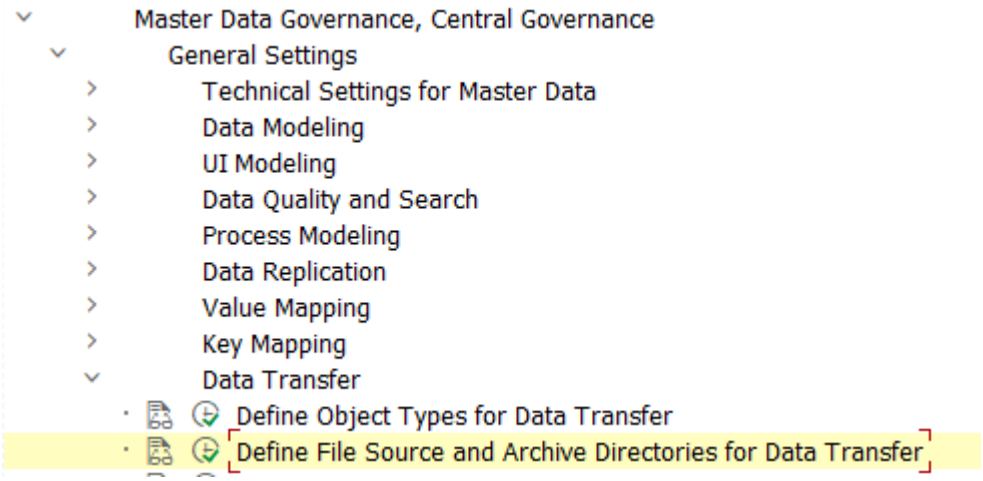
e.g.: /usr/sap/<systemID>/MDG_DATA_ARCHIVE/<FILENAME>

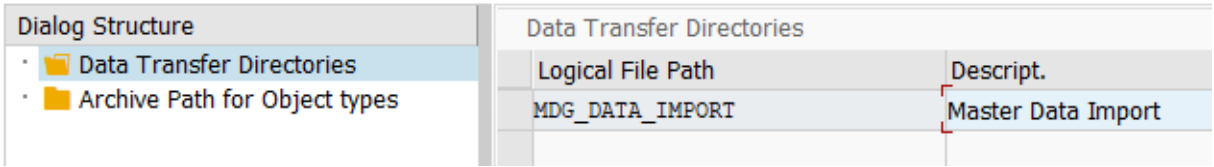


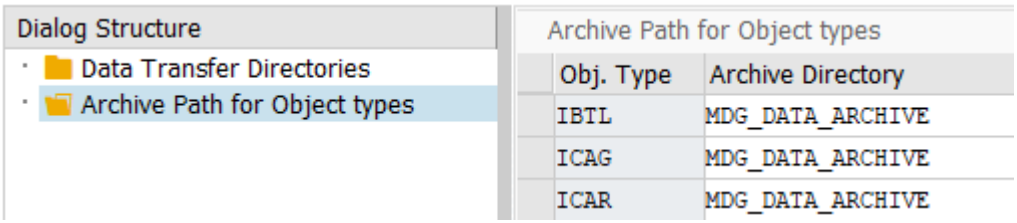




Using transaction MDGIMG configure the usage of these directories:







Using MDMGX and Data Import all reference data object types which do not belong to a hierarchy can be extracted. See "Reference Data Management for SAP MDG - Functional Documentation" for usage details and the next section for details about the configuration and load of a product hierarchy.

3.12 Configure Product Hierarchy

If the product hierarchy is within the scope of maintained reference data object types additional steps need to be performed on the MDG RDM system as well as on the SAP business application system.

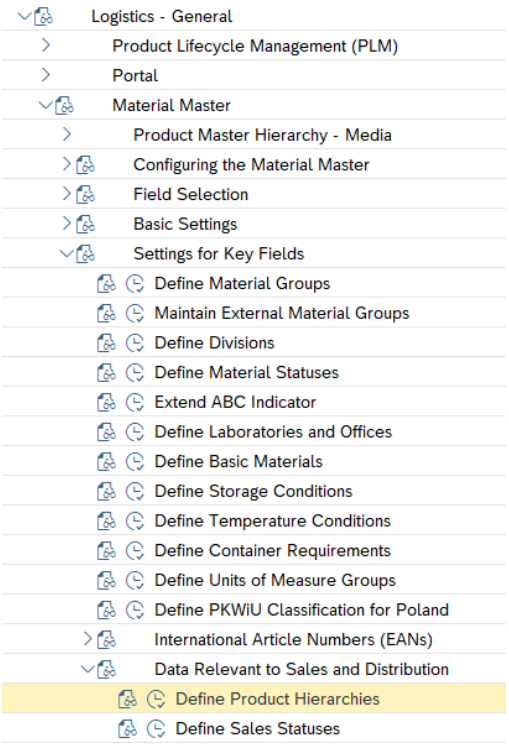
On the MDG RDM System the following activities need to be performed:

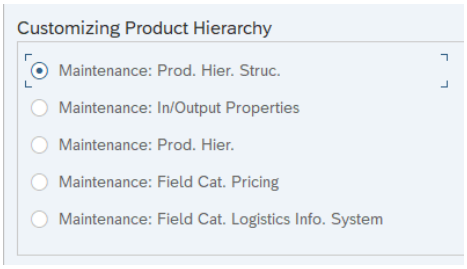
- Verify the product hierarchy usage in your business applications
- Configure the product hierarchy usage in RDM
- Check Interlocking
- Define Edition
- Creation of a Product Hierarchy Name
- Maintenance of number ranges
- Data Import (File Upload) after Data Export from the business application system

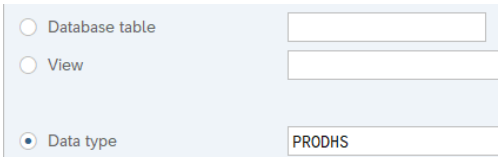
3.12.1 Verify the product hierarchy usage in your business applications

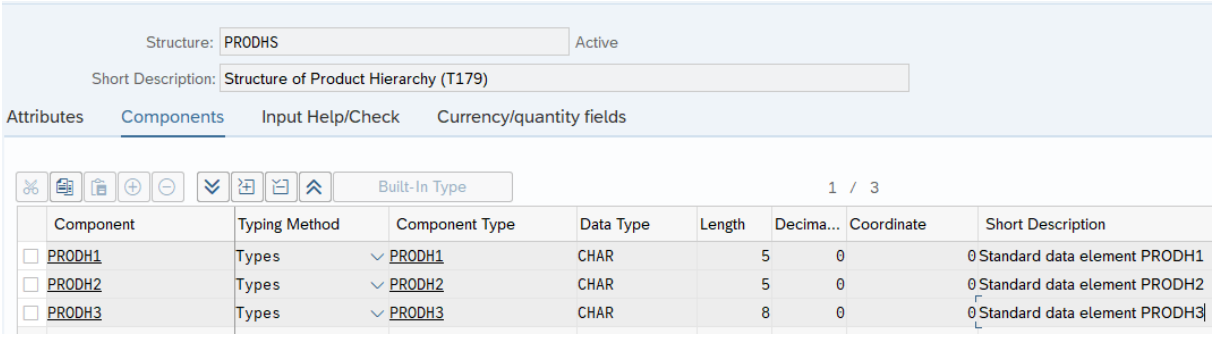
The RDM standard delivery assumes an SAP standard configuration of three levels and number ranges have to be maintained according to the Product Hierarchy Set Up in the receiving business applications. The Set Up of the Product Hierarchy in a SAP business application is done through the definition of structure PRODHS.

To check the configuration use transaction SPRO and check the following entries:


Logistics - General
Product Lifecycle Management (PLM)
Portal
Material Master
Product Master Hierarchy - Media
Configuring the Material Master
Field Selection
Basic Settings
Settings for Key Fields
Define Material Groups
Maintain External Material Groups
Define Divisions
Define Material Statuses
Extend ABC Indicator
Define Laboratories and Offices
Define Basic Materials
Define Storage Conditions
Define Temperature Conditions
Define Container Requirements
Define Units of Measure Groups
Define PKWiU Classification for Poland
International Article Numbers (EANs)
Data Relevant to Sales and Distribution
Define Product Hierarchies
Define Sales Statuses


Customizing Product Hierarchy
Maintenance: Prod. Hier. Struc.
Maintenance: In/Output Properties
Maintenance: Prod. Hier.
Maintenance: Field Cat. Pricing
Maintenance: Field Cat. Logistics Info. System


Database table
View
Data type
PRODHS


Structure: PRODHS
Active
Short Description: Structure of Product Hierarchy (T179)
Attributes
Components
Input Help/Check
Currency/quantity fields
Built-In Type
1 / 3
Component
Typing Method
Component Type
Data Type
Length
Decima...
Coordinate
Short Description
PRODH1 Types PRODH1 CHAR 5 0 0 Standard data element PRODH1
PRODH2 Types PRODH2 CHAR 5 0 0 Standard data element PRODH2
PRODH3 Types PRODH3 CHAR 8 0 0 Standard data element PRODH3

The figure above shows the SAP standard configuration. Please refer to the following SAP note for more information: [1070939 - Consulting: Changing the product hierarchy structure PRODHS - SAP for Me].

3.12.2 Configure the product hierarchy usage in RDM

Use transaction /ITR/RDMIMG:

Configure RDM Product Hierarchy level descriptions:

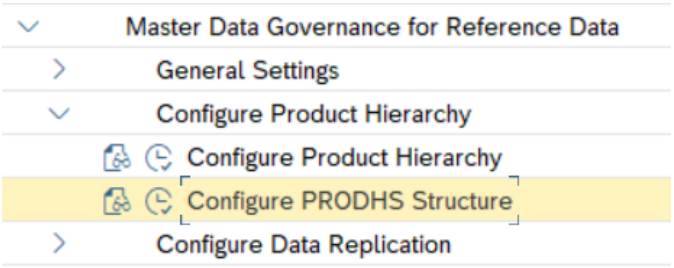
Example for a 3-level product hierarchy:

Prod.H.Lvl	Description (medium text)
1	Division
2	Brand
3	Segment

Note: In the standard delivery all levels are represented by Product Hierarchy Nodes. Optional configuration for Product Hierarchy Sub Nodes is possible and would be activated for the next level which is not defined in the configuration shown above. Example: a configuration of Division and Brand means that a level for "Segment" can be represented by Product Hierarchy Nodes. This might be useful If additional attributes should be defined for the "leaf nodes" of the Product Hierarchy.

For more levels Type, Series and Model might make sense.

After this, configure the RDM Product Hierarchy external number length:



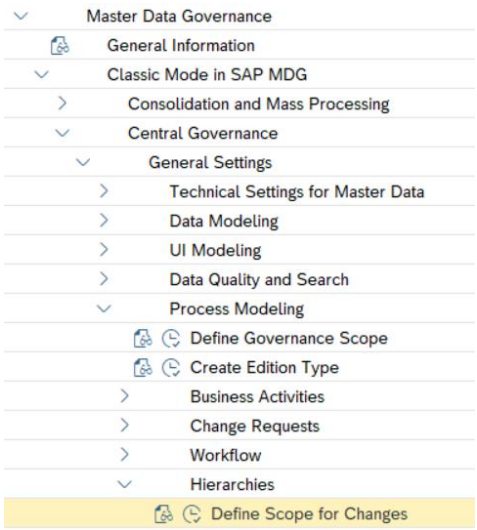
Example for a 3-level product hierarchy:

PRODHS fields	Length
PRODH1	000005
PRODH2	000005
PRODH3	000008

For a 7-level hierarchy probably 000002, 000002, 000002, 000002, 000003, 000003, 000004 might make sense. It is possible to define up to 9 levels and the number should add up to not more than 18 (0000018) characters.

3.12.3 Check Interlocking

Execute transaction MDGIMG: (Classic Mode in SAP MDG) -> Central Governance -> Process Modelling -> Hierarchies -> Define Scope for Changes -> Data Model I1 -> Scope for Changes -> Hierarchy Type: Product Hierarchy: PRODH -> Interlocking. This needs to be defined as "Strict" and shall not be changed as inconsistencies in the Product Hierarchy might occur.

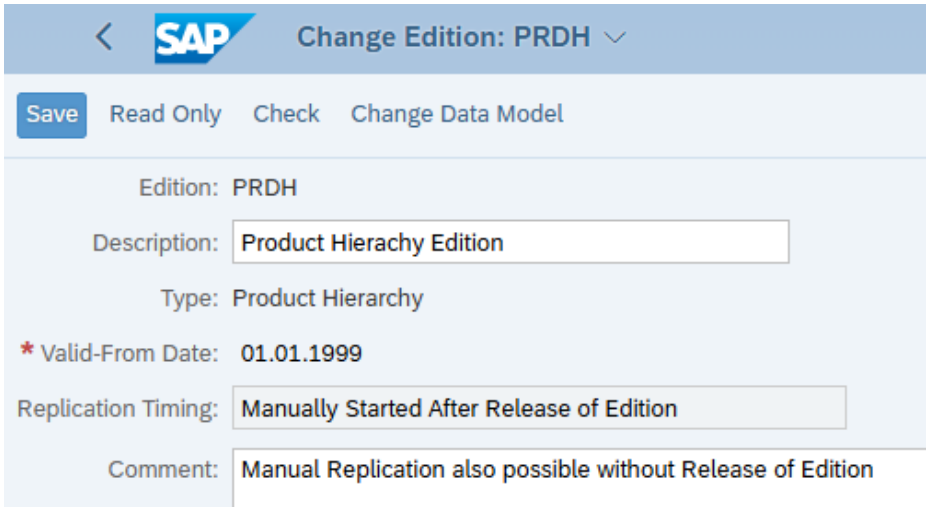


Hierarchy Type	Description	Interlocking
PRODH	Product Hierarchy	Strict

3.12.4 Define Edition

For the definition of an Edition start the Fiori Launchpad and select Analysis of Editions -> Create. Use Edition Type "Product Hierarchy (I1_PRODH)" and define the Data Replication Timing "Manually started after Release of Edition"

Example:



Note: Parallel editions are currently not supported.

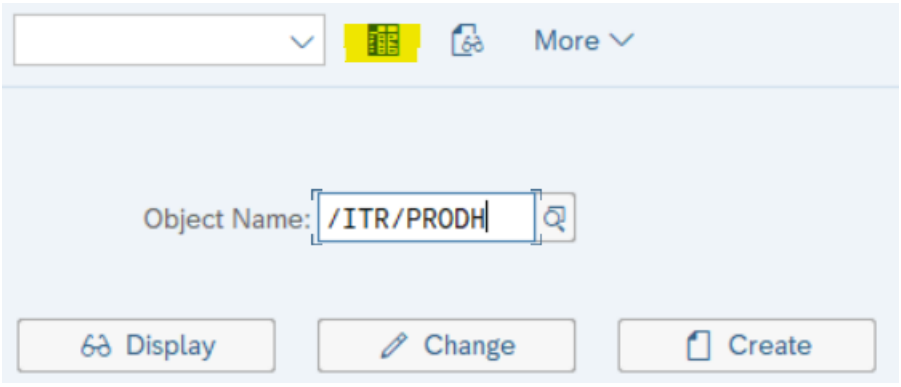
3.12.5 Creation of a Product Hierarchy Name

The creation of the Product Hierarchy Name is done through a change request process. Start the Fiori Launchpad and select Change Requests -> Sales -> Product Hierarchy Name -> New -> Change Request Type: IPN01 (Create Product Hierarchy Name). Select your edition and continue.

The RDM system on default uses the Product Hierarchy Name ID "ProdHrchy". Submit and verify that the change request is finalized automatically.

3.12.6 Maintenance of number ranges

The maintenance of number ranges starts with definition of the "From Number" and the "To Number" for each level of the Product Hierarchy Nodes and the (optional) Product Hierarchy Sub Nodes. After the initial load of the Product Hierarchy also the Number Range Status needs to be maintained before new Nodes or Sub Nodes can be created. Use transaction SNRO (Object: /ITR/PRODH) and choose "Interval Editing" for the maintenance process.



Example:

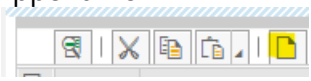
Number Range No.	From No.	To Number	NR Status
01	000000000015000000	000000000019999999	15000000
02	000000000025000000	000000000029999999	25000000
03	000000000035000000	000000000039999999	35000000
04	000000000045000000	000000000049999999	45000000
05	000000000055000000	000000000059999999	55000000
06	000000000065000000	000000000069999999	65000000

Note: It is recommended to use the same or a similar number range. The idea here is that the first digit is representing the product hierarchy level and the creation of nodes in the user interface will start with 15000000 (for the first level). Data loaded from external sources (see next section) will start with 10000000 creating a number range from 10000000 to 14999999 for uploads (into the first level) and in general for all levels such a number range should be used for the uploads. Please find more details for the extraction and upload of product hierarchy nodes below.

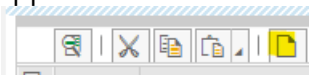
3.12.7 Data Export from business application system

For the data export from the SAP business application system execute transaction SA38 and choose report /ITR/MDG_PRODH_EXPORT to extract the product hierarchy using the defined number ranges which have been configured for MDG RDM. Choose the following parameters:

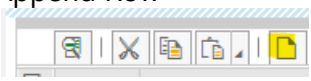
- Product Hierarchy Name: "ProdHrchy"
- Delimiter: ";"
- Output Folder on local desktop: <directory> (any directory which can be accessed by the user executing the report)
- Append Row



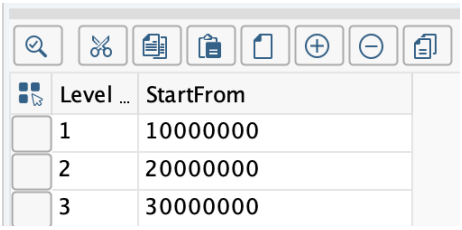
  - Level: 1; <StartFrom> (any number which fits to the number range defined above for range number 01; e.g. 10000000 for the initial load)
- Append Row



  - Level: 2; <StartFrom> (any number which fits to the number range defined above for range number 02; e.g. 20000000 for the initial load)
- Append Row



  - Level: 3; <StartFrom> (any number which fits to the number range defined above for range number 03; e.g. 30000000 for the initial load)



Level ...	StartFrom
1	10000000
2	20000000
3	30000000

For a five level hierarchy e.g. the following entries might be a good example:

- 1: 10000000
- 2: 20000000
- 3: 30000000
- 4: 40000000
- 5: 50000000

3.12.8 Data Import (File Upload)

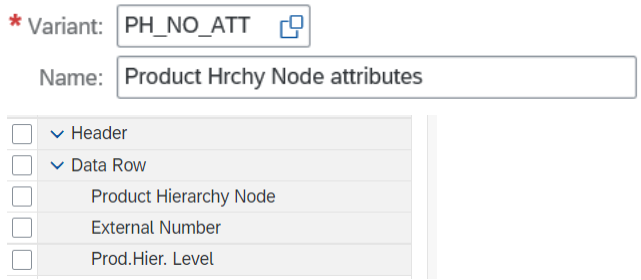
After the Data Export the created files have to be loaded to the MDG RDM system using File Upload. This should be done in the following sequence for a product hierarchy with 3 levels:

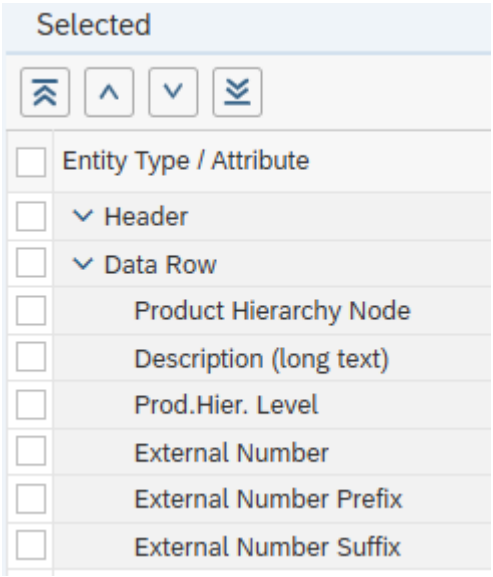
- Level 1 to 3: attributes and texts
- Level 1: hierarchy assignments of level 1 nodes to the product hierarchy name
- Level 2: hierarchy assignments of level 2 nodes to level 1 nodes
- Level 3: hierarchy assignments of level 3 nodes to level 2 nodes

Please verify that all numbers have been generated according to the number ranges defined.

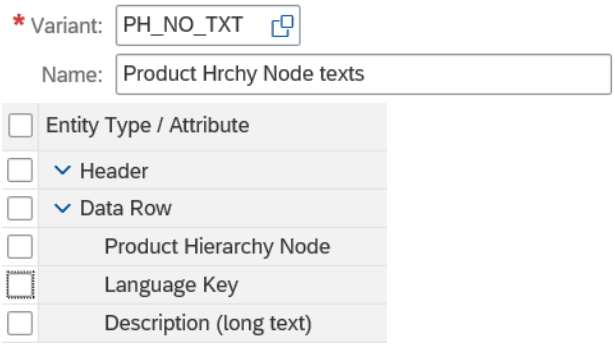
For this, the following upload variants have to be defined:

- Level 1 attributes and texts (PH_NO_ATT and PH_NO_TXT)
  - Entity Type: Product Hierarchy Node
  - Attributes Data Row: Product Hierarchy Node, External Number, Prod.Hier.Level





- Texts Data Row: Product Hierarchy Node, Language Key, Description (long text)



- Level 2 (and higher) attributes and texts
  - Same Entity Type and variants as for Level 1 Nodes

- Level 1 hierarchy assignments of level 1 nodes to the product hierarchy name
  - Entity Type: Product Hierarchy Node
  - Higher-level Node: Product Hierarchy Name
  - Lower-level Node: Product Hierarchy Node

- Level 2 (and higher) hierarchy assignments of level 2 nodes to level 1 nodes
  - Entity Type: Product Hierarchy Node
  - Higher-level Node: Product Hierarchy Name, Product Hierarchy Node
  - Lower-level Node: Product Hierarchy Node

Use the variants above to load the data with

- Upload Mode: "Overwrite or Add"
- Conversion: "Execute Conversion"
- File System / File Name: <upload directory>
- Separator: "Semicolon"
- Comment Row: "*"
- Change Request Type: "Load Prod. Hier. And activate"
- Description: <any>

After each File Upload check the objects using the search application for the object: Change Requests -> Processing -> Sales -> Product Hierarchy Node.

After this the maintenance of the product hierarchy can start and the next created nodes (using the Single Object Maintenance) will get the next number. Nodes which are created by File Upload will get the numbers which are specified in the files. Aligned with the examples used above make sure that you use numbers from the following number ranges for the creation of new nodes (example for a three level hierarchy):

- 1: 10000000 - 14999999
- 2: 20000000 - 24999999
- 3: 30000000 – 34999999

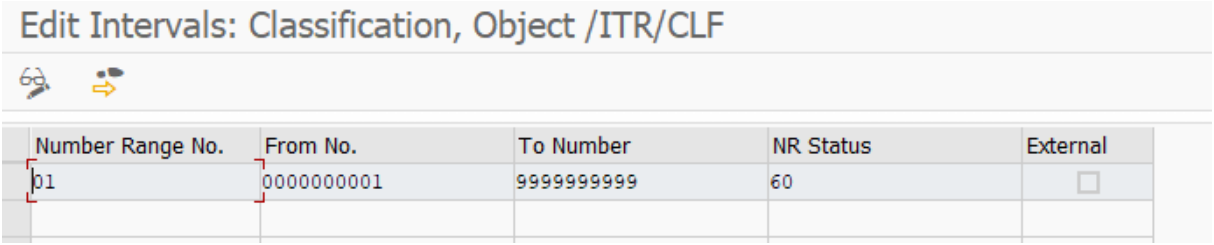
3.12.9 Enabling 'Add to Change Request' Functionality

To enable the "Add to Change Request" functionality in the user interface, the corresponding SAP Note 3668887 must be implemented in the system. This note ensures that the required backend functionality is available and prevents dumps during processing. This functionality ensures that the description of existing hierarchy nodes can be changed without the need to navigate to the change request itself. Mark the node, add it to the change request and change the description. Without this functionality you need to navigate to the change request, and add the hierarchy node manually before you are able to change the description.

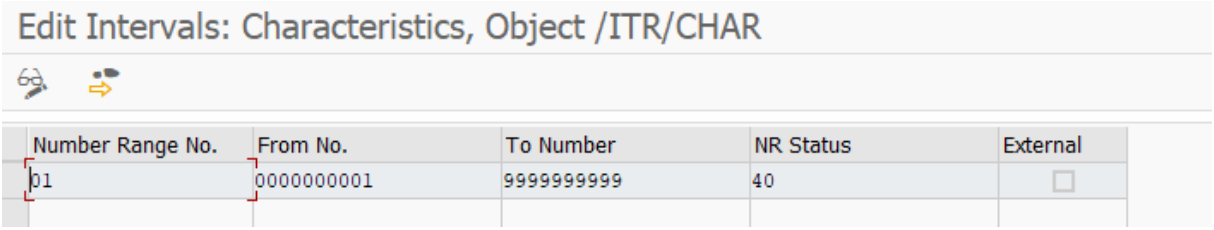
3.13 Configure Classification

Classes and Characteristics are implemented with an internal numbering. This means that number ranges need to be maintained using the transaction SNRO (or SNUM). Examples:

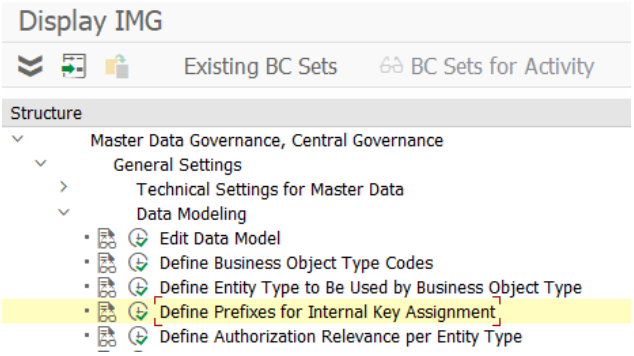
Classes:

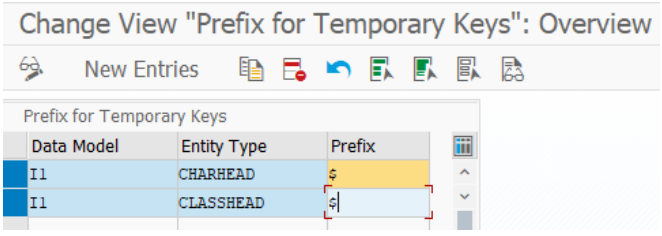


Characteristics:



Additional a prefix for the temporary keys needs to be defined:





3.14 Configure Public Holiday

If the Public Holiday is within the scope of maintained reference data object types additional steps need to be performed on the MDG RDM system for the correct data import and number range settings.

Prerequisite:

- Data Export from SAP S/4 has been executed

Step 1: Set Number Range

- Transaction: SM30
  - Table/View: /ITR/NRO_FTGID
  - Maintain
  - Entity Type: FTGID
  - Public hol. ID: 000

Step 2: Data Import

- Execute Data Import or File Upload
- Execute Search (with Maximum Number of Hits = 5000) and identify the last number

Step 3: Set Number Range Status

- Transaction SM30
  - Table/View: /ITR/NRO_FTGID
  - Maintain
  - Entity Type: FTGID
  - Set Public hol. ID to the last number identified from step 2

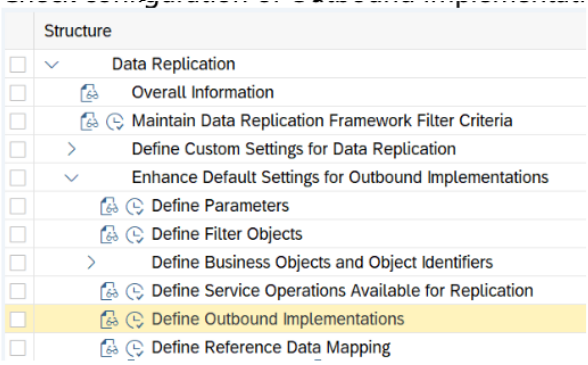
3.15 Configure Initial Load of Exchange Rates

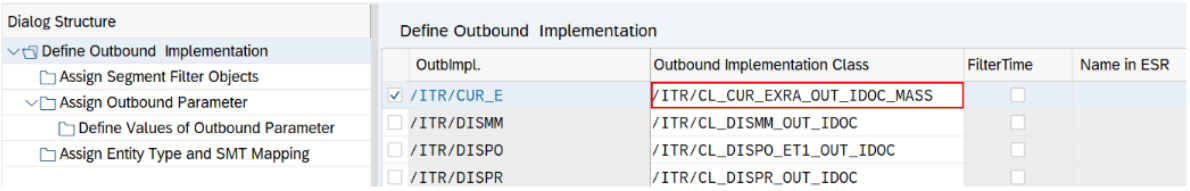
For customers processing a large volume of exchange rates during an initial mass load, a potential performance issue has been identified in the standard system setup, which can lead to application dumps depending on the memory size of the system to be loaded.

To resolve this issue and ensure stable performance during the initial mass load of exchange rates, you must apply a specific configuration with regards to the IDoc packaging size for the relevant outbound implementation.

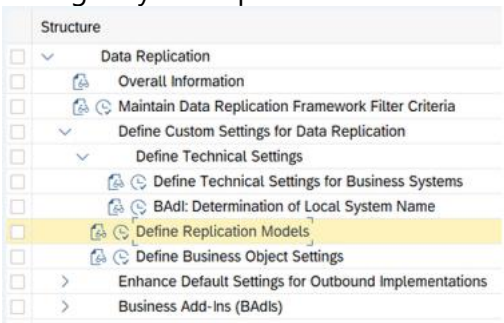
Start transaction DRFIMG.

Check or adjust the configuration of the Outbound Implementation.

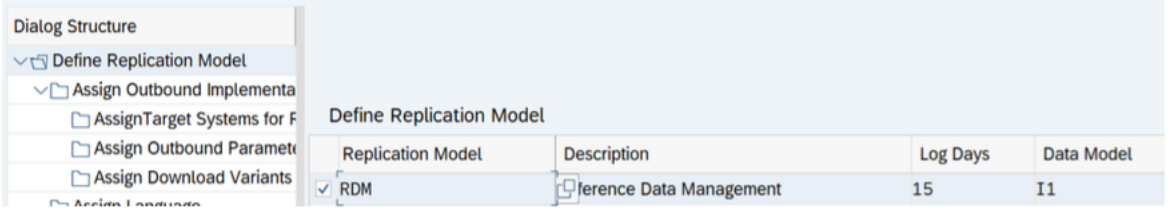




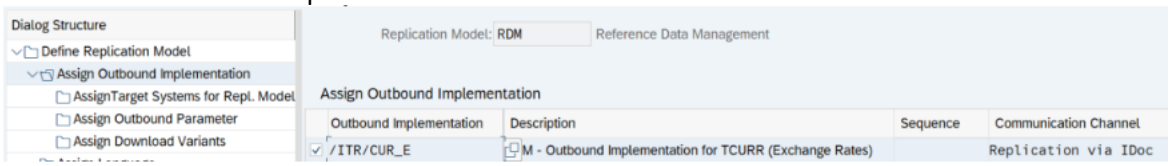
Configure your Replication Model



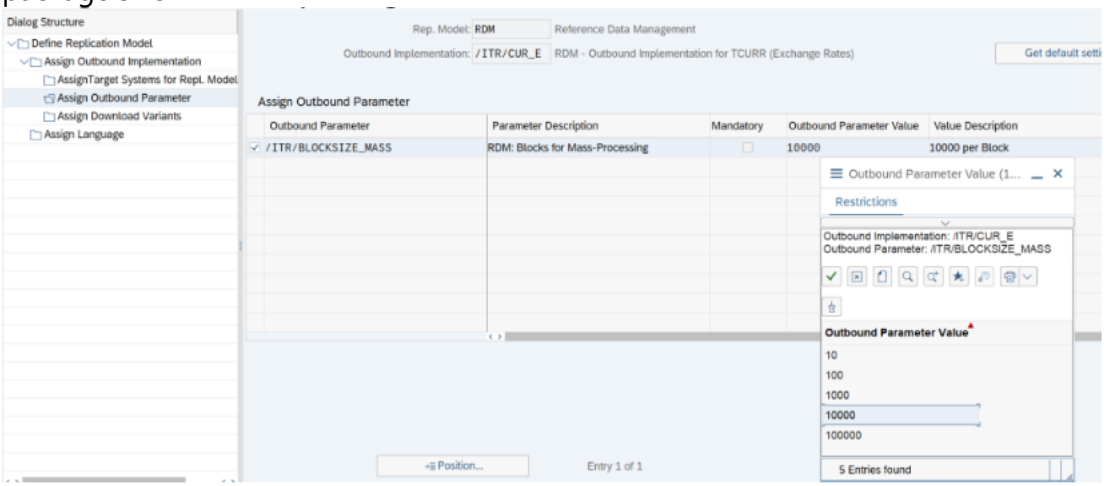
Select your Replication Model



Select the outbound implementation



Within the configuration settings for the outbound implementation, configure a suitable package size



It is recommended to set the IDoc package size such that the system creates IDocs in packages of 10,000. This package size ensures stable performance during the initial mass load and that the receiving system can process all packages efficiently. Please adjust if this setting does not meet the requirements of your SAP system. You might use higher or lower numbers depending on your system sizing.

3.16 Set Up Local Staging Area

The configuration of the Local Staging Area in the SAP receivers can be done by importing BC Set /ITR/RDM_STAGING_<n> "MDG RDM Staging Settings" (see: 3.3 "Activate Business Configuration Sets"). Make sure that you have to do this for each SAP receiver.

Also make sure that you define users with role /ITR/MDG_STAGING_<n> "User for Local Staging Area". See section 3.4 "Configure User Roles and Authorizations".

Note: any user, who wants to run the transaction /ITR/STAGING needs write-access to the transport directory of the corresponding ERP system. In most cases, the directory usually is "/usr/sap/trans", but could also be called differently. It might therefore be necessary to check the folder path with SAP basis and provide access for the user.

3.17 Solution Manager Integration

Please refer to the separate document "MDF Solution Manager Integration".

In order to obtain more information about these and other project specific enhancements contact: support@itego.de - Subject: "RDM Solution Manager Integration".

3.18 Project Specific Enhancements

MDG RDM offers functionalities which are not activated and not supported in the standard delivery but can be activated based on project specific requirements. Examples are:

- Status Net (Controlled maintenance of an object status)
- Object Deletion (Governed deletion of objects)

In order to obtain more information about these and other project specific enhancements contact: support@itego.de - Subject: "RDM Project Specific Enhancements".

4 Configuration for Reference Data Harmonization

4.1 Activate Business Configuration Sets

4.1.1 SAP MDG: BC Sets

4.1.1.1 Software Component ITG

In order to get predefined Configuration Groups for Reference Data Harmonization please also consider to activate the following BC Set (please be aware that these are needed only when you use the MDG system for the synchronization of reference data types which are not covered in your Reference Data Governance scenarios):

- /ITR/RDH_CONFIG_GROUPS_<n> RDH Configuration Groups

4.2 Configure User Roles and Authorizations

4.2.1 Reference Data Harmonization – Sender

The following roles are delivered for Reference Data Harmonization (on the sender system which might be in most case SAP MDG) and define which actions are allowed for which user.

- /ITR/ITEGO_MDG_RDH_DISP_<n>S Display Functions in the Sender-System
- /ITR/ITEGO_MDG_RDH_BUSINES_<n>S Business Functions in the Sender-System
- /ITR/ITEGO_MDG_RDH_EXPERT_<n>S Expert Functions in the Sender-System

4.2.2 Reference Data Harmonization – Receiver

The following roles are delivered for Reference Data Harmonization (on the receiver system which might be the SAP MDG system when data is consolidated in a first step in SAP MDG) and define which actions are allowed for which user.

- /ITR/ITEGO_MDG_RDH_DISP_<n>E Display Functions in the Receiver-System
- /ITR/ITEGO_MDG_RDH_BUSINES_<n>E Business Functions in the Receiver-System
- /ITR/ITEGO_MDG_RDH_EXPERT_<n>E Expert Functions in the Receiver-System

5 Configuration Examples

5.1 Data Replication based on SAP ALE

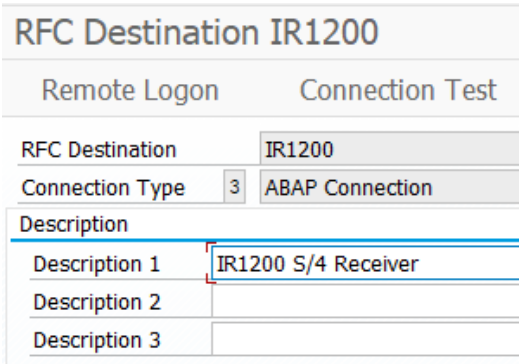
Most RDM objects will be replicated using a generic message type. This section shows how to set this up based on two systems:

- RDM for MDG (sender): System IH1, client 100
- SAP Business System (receiver): System IR1, client 200

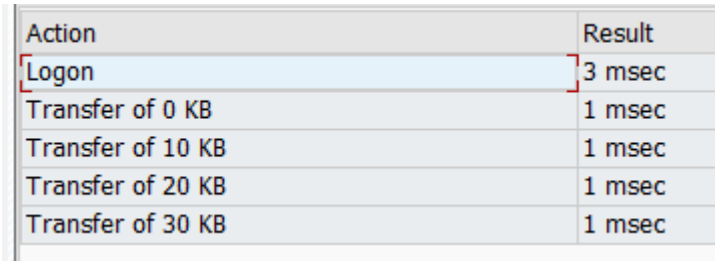
Please adjust the examples below to your own system landscape.

5.1.1 Prerequisite: RFC Destination

System IH1 100: Check or create a RFC destination to system IR1 200 using transaction SM59:



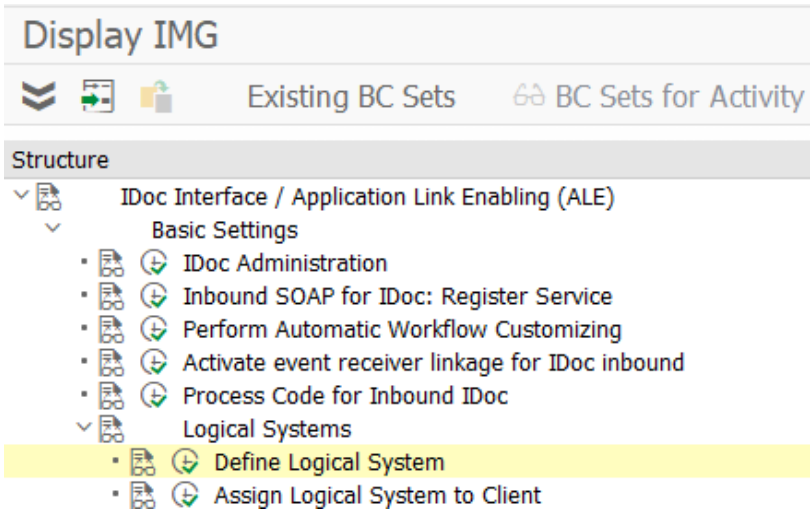
Connection Test should be successful

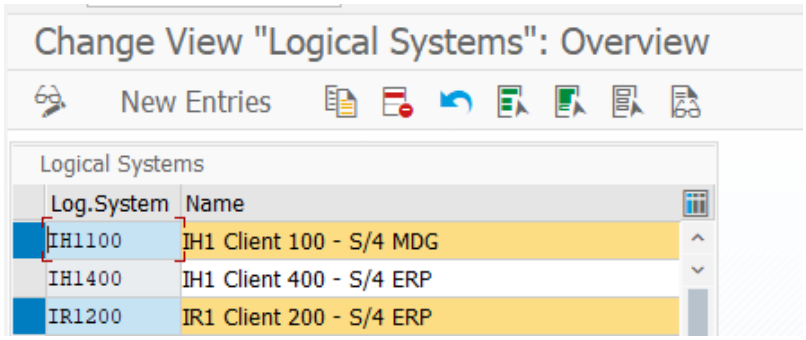


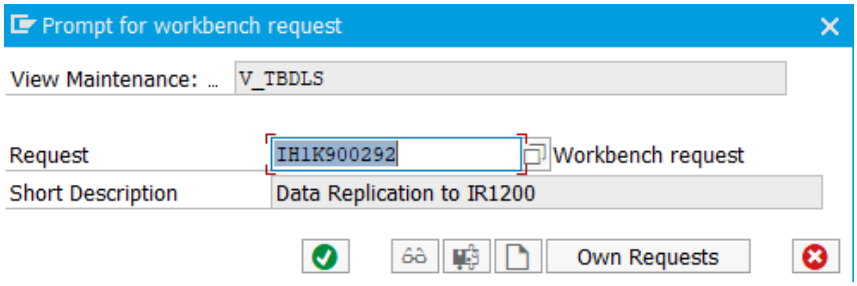
Action	Result
Logon	3 msec
Transfer of 0 KB	1 msec
Transfer of 10 KB	1 msec
Transfer of 20 KB	1 msec
Transfer of 30 KB	1 msec

5.1.2 Define Logical System and Check or Create Business System

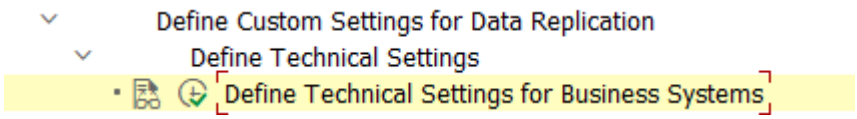
System IH1 100, transaction SALE:

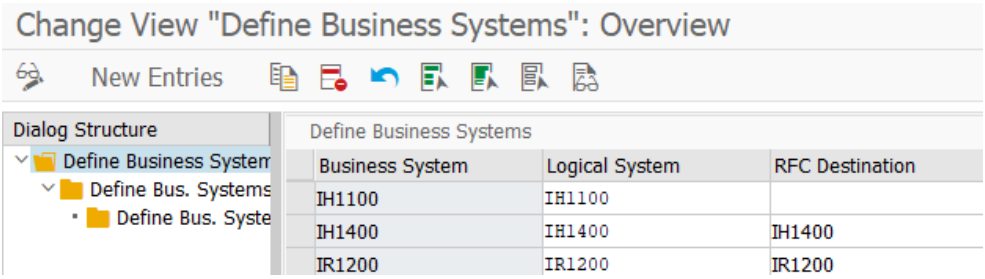






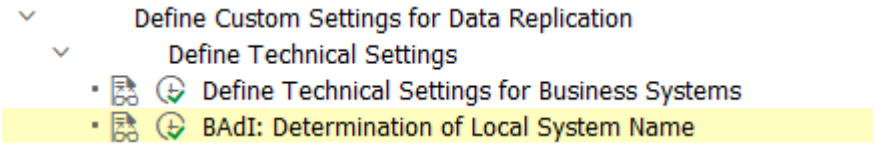
Check or Create Business System: System IH1 100, transaction DRFIMG



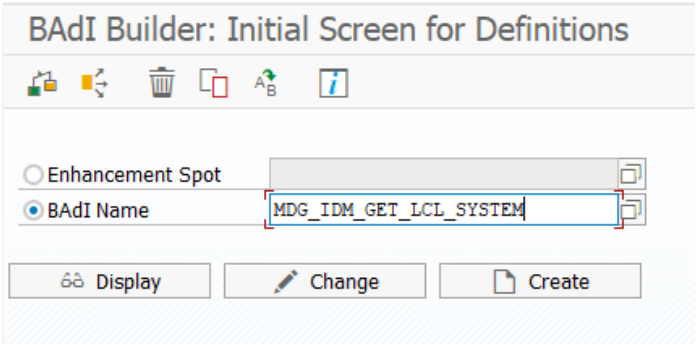


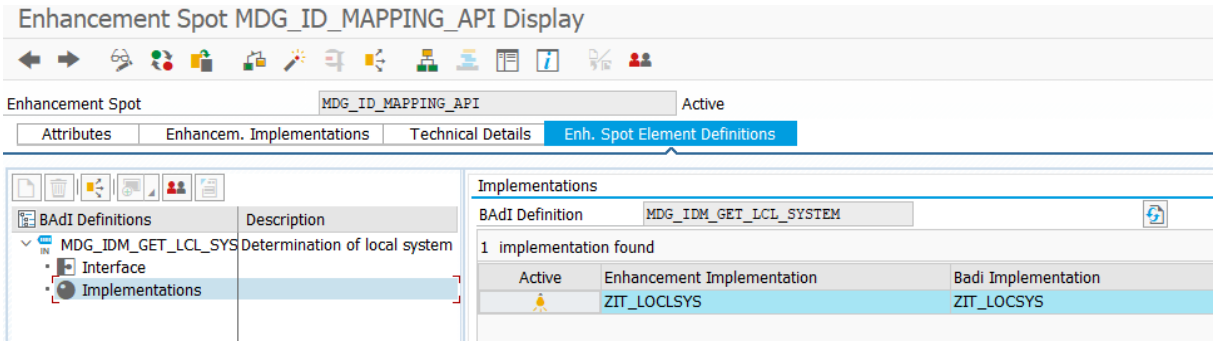
5.1.3 Check or Create BAdI for determination of local system name

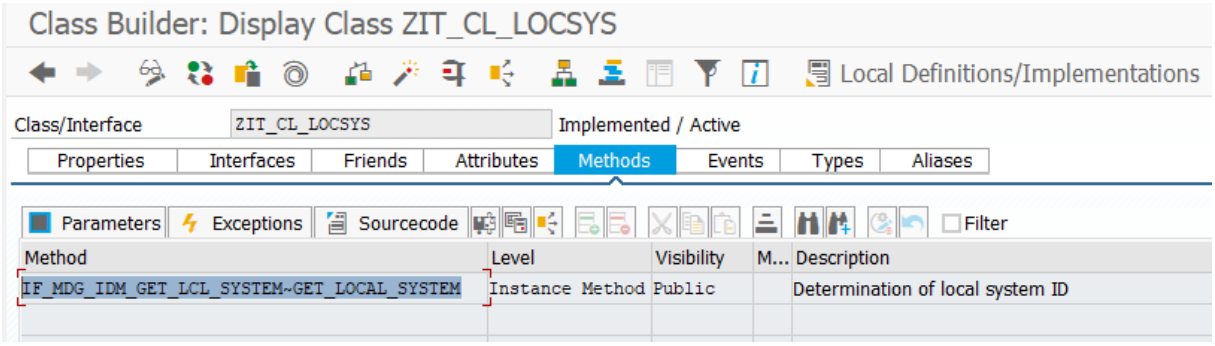
System IH1 100, transaction DRFIMG:



Or use transaction SE18, BAdI Name: MDG_IDM_GET_LCL_SYSTEM



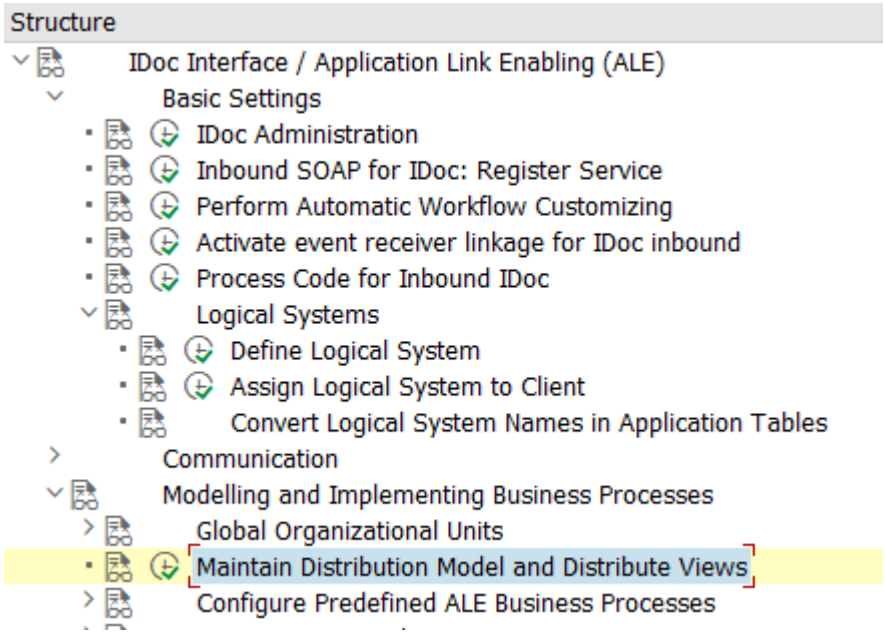




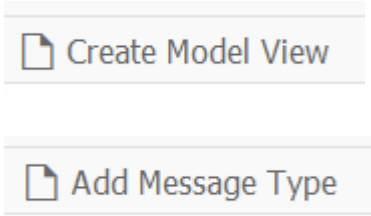

```
method IF_MDG_IDM_GET_LCL_SYSTEM~GET_LOCAL_SYSTEM.
*! This method determines the local business system via the ALE logical sys
tem assigned to it.
* If no business system is maintained, it returns an empty value.

  DATA:
    lv_own_logical_system TYPE logsys,
    ls_bs_tech            TYPE mdg_s_bus_sys_tech,
    lv_not_found          TYPE boole_d.

  CALL FUNCTION 'OWN_LOGICAL_SYSTEM_GET'
    IMPORTING
      own_logical_system             = lv_own_logical_system
    EXCEPTIONS
      own_logical_system_not_defined = 1
      OTHERS                         = 2.

  cl_mdg_bs_access_cust_data=>select_bs_data_for_logsys(
    EXPORTING
      iv_logsys = lv_own_logical_system
    IMPORTING
     es_bs_tech   = ls_bs_tech
     ev_not_found = lv_not_found ).

  IF lv_not_found = abap_false.
    ev_local_system = ls_bs_tech-business_system.
  ENDIF.

  endmethod.
```

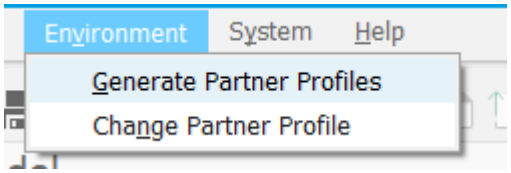
5.1.4 Define Distribution Model

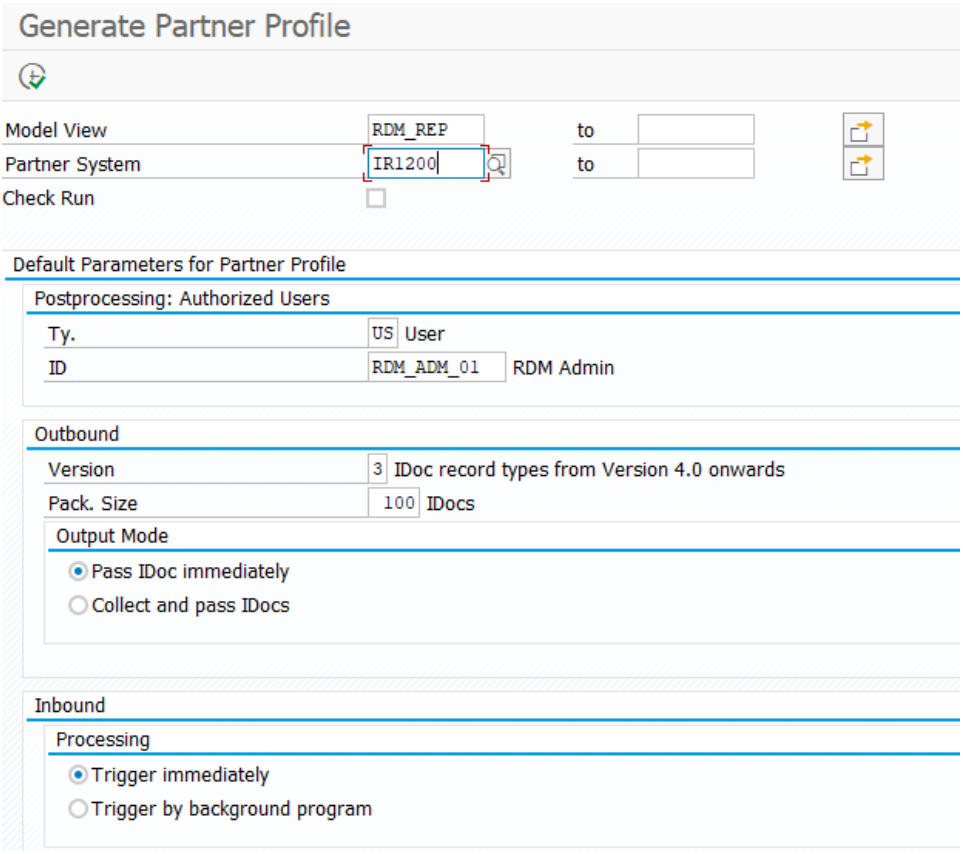
Use transaction SALE (or BD64):

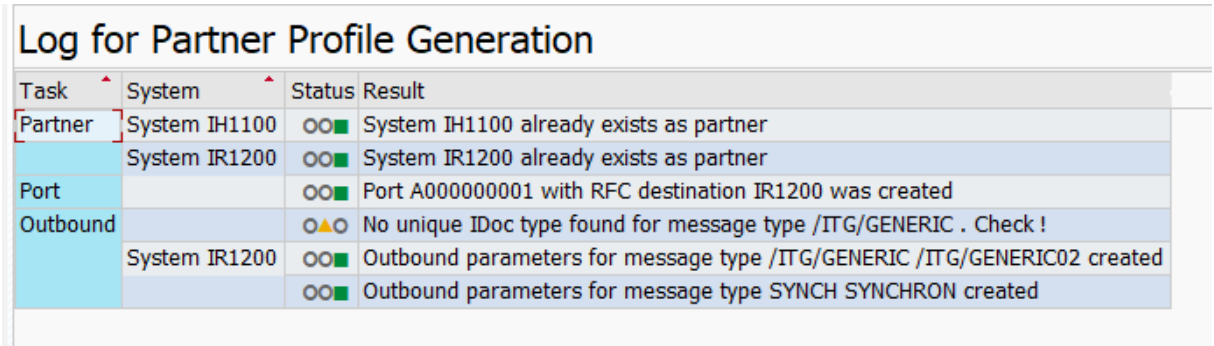


Create Model View and Add Message Type

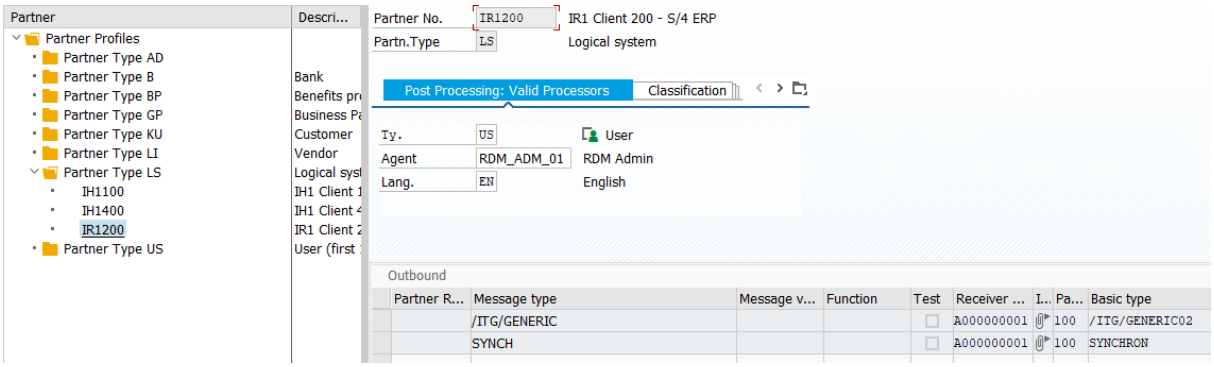




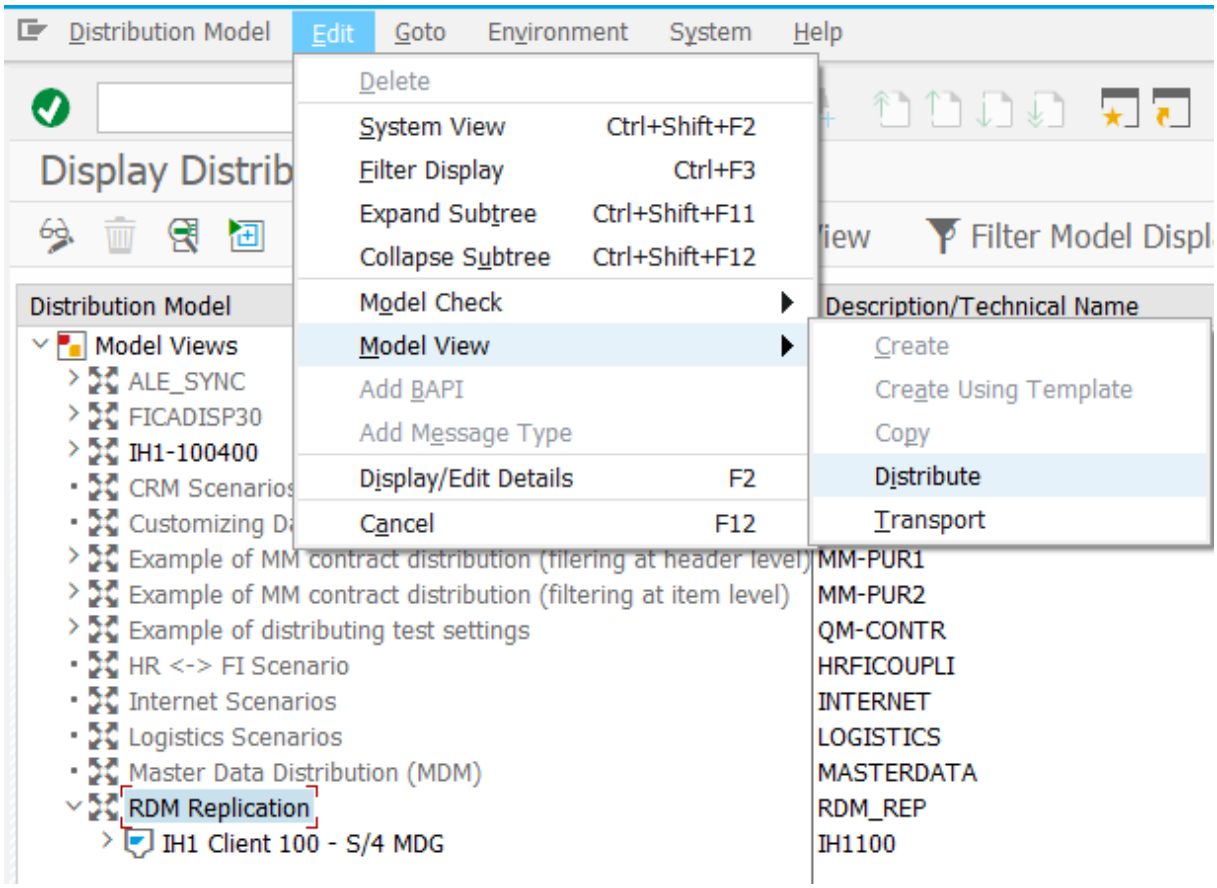


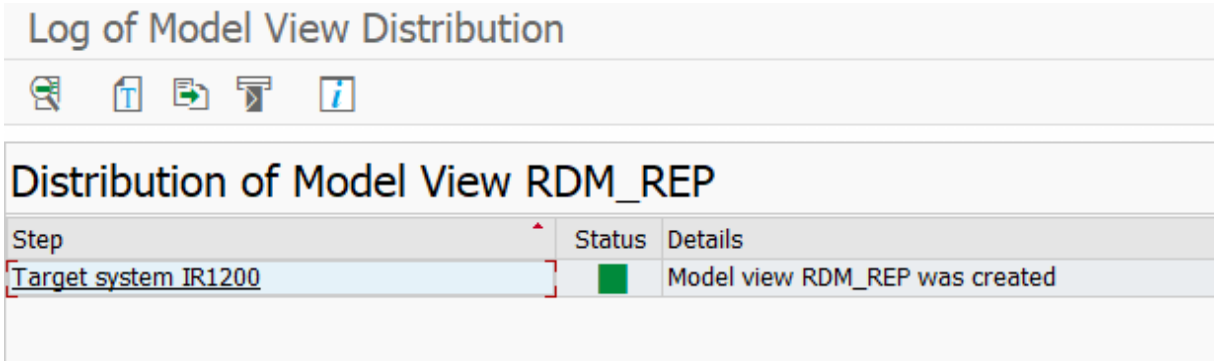


Check in transaction WE20:



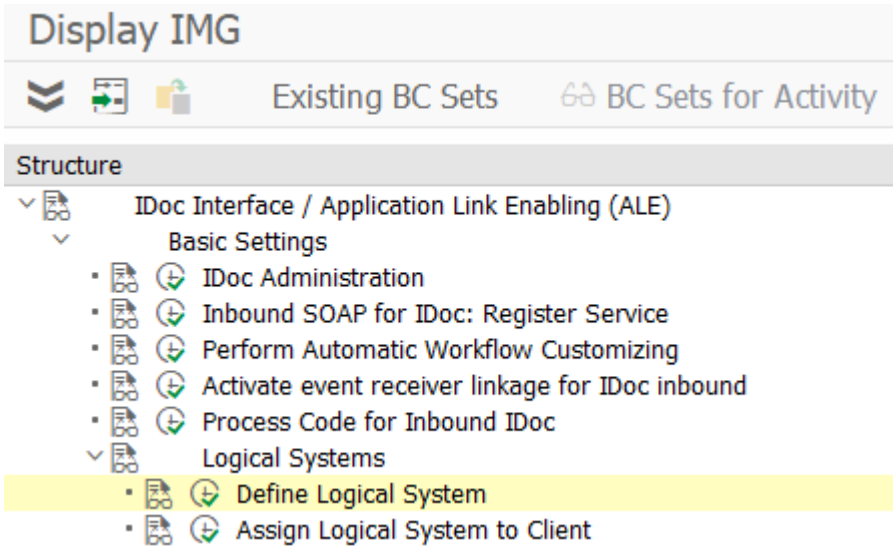
BD64: Distribute Model View to receiver (IR1200)

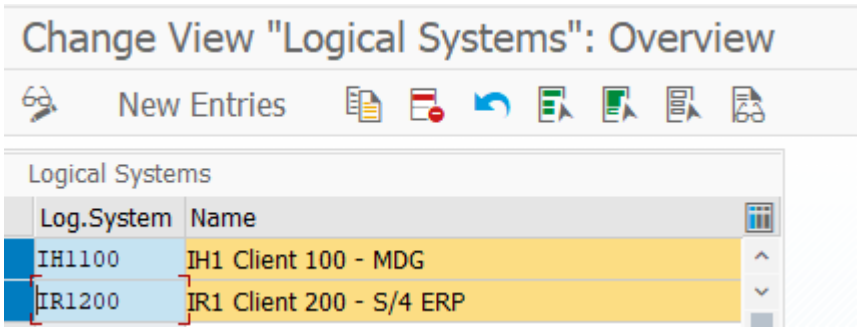


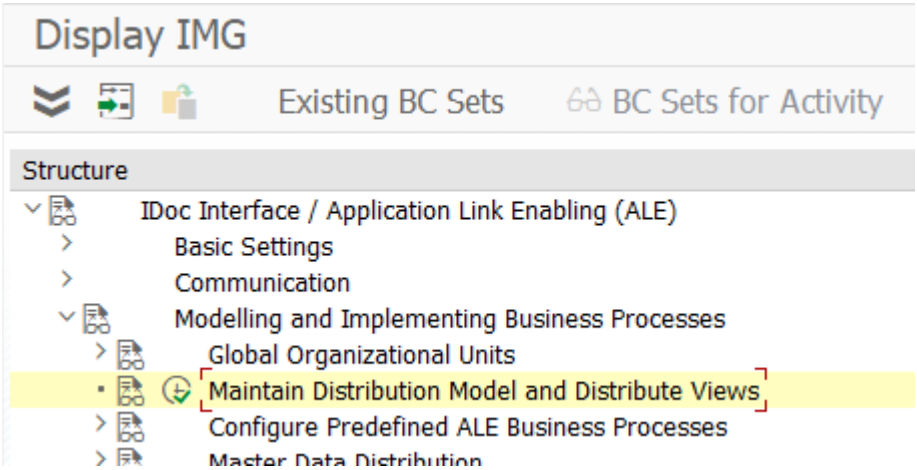


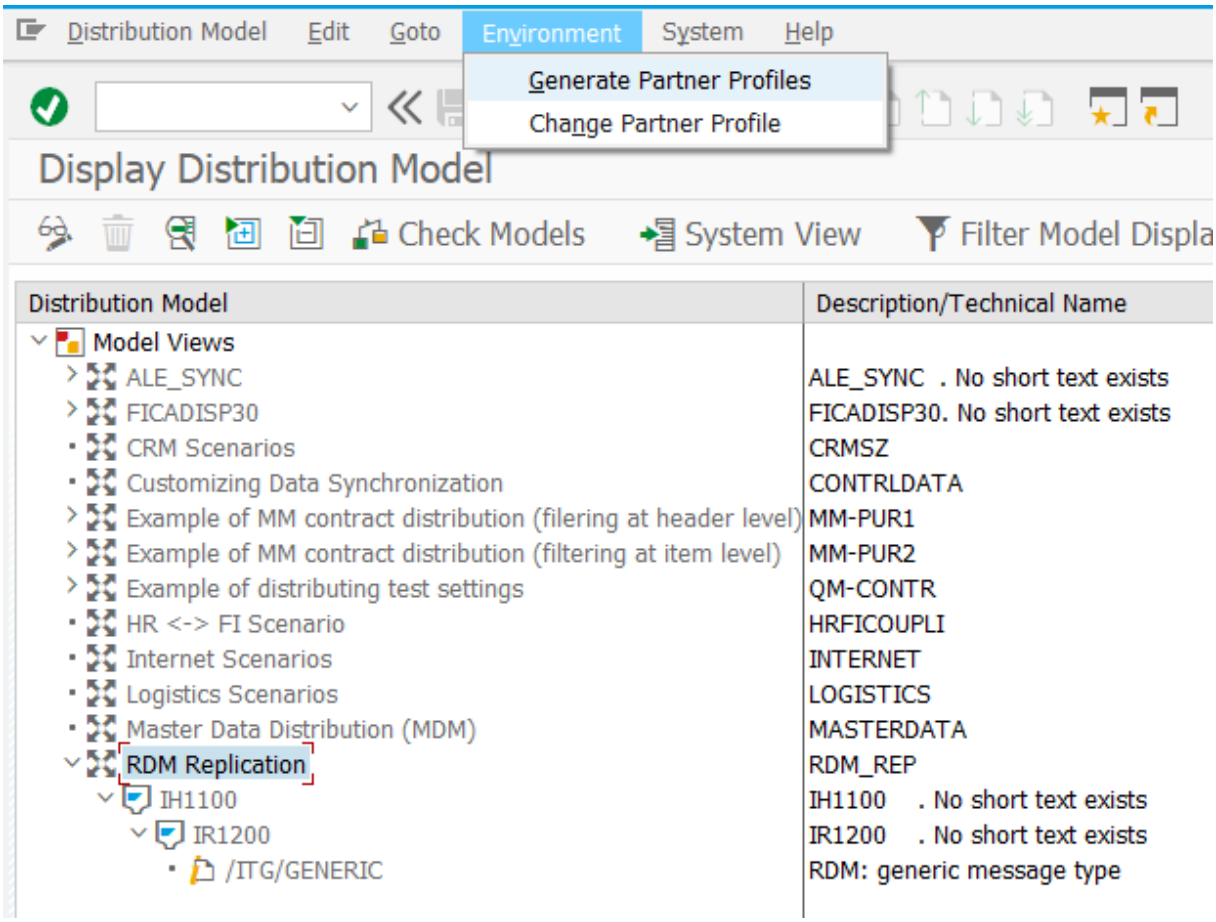
Log On to system IR1200 and execute transaction SALE:

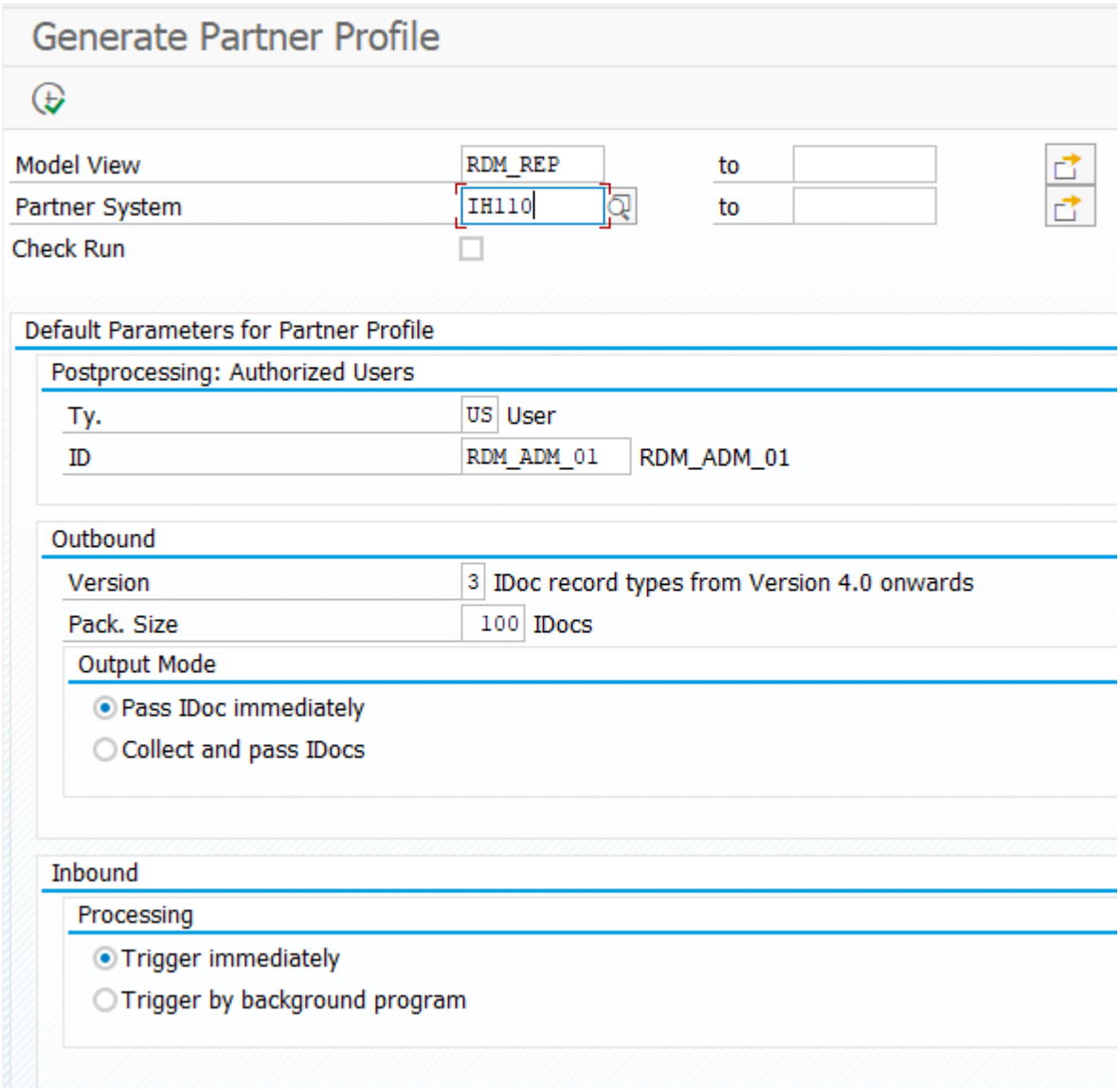
Check Logical Systems



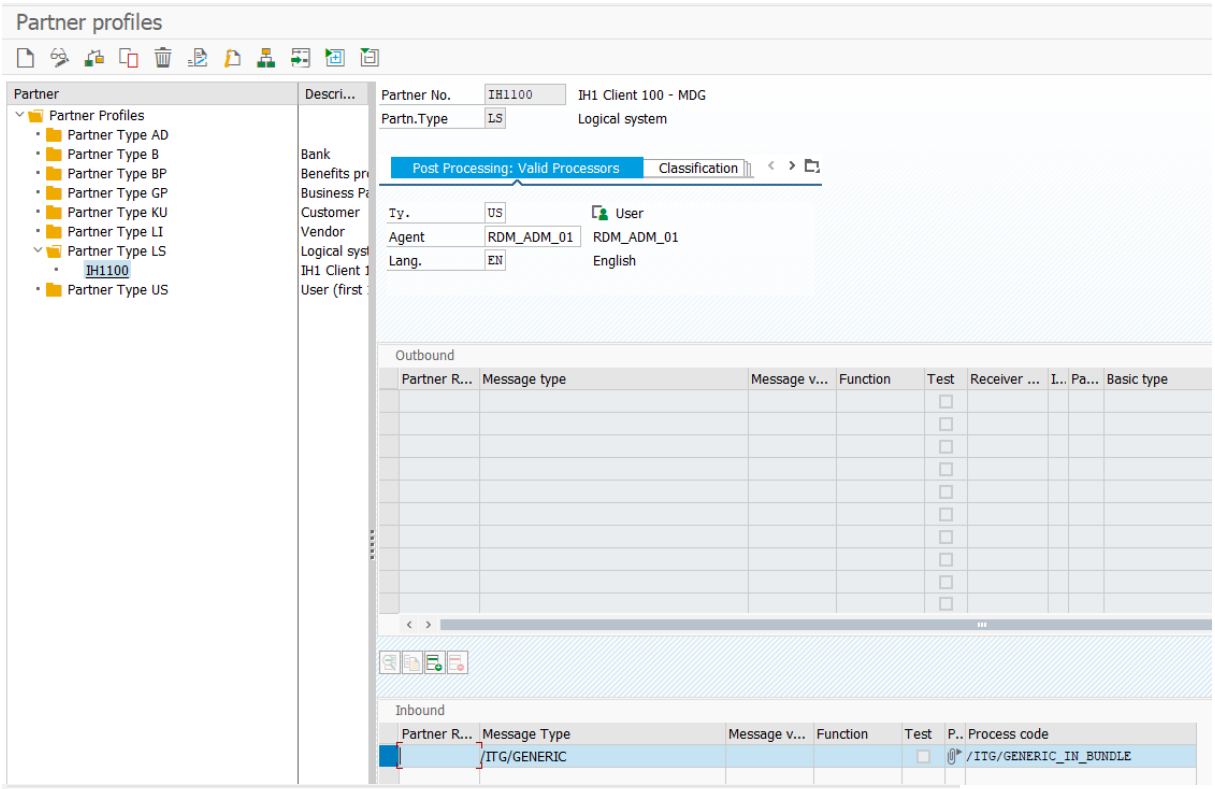


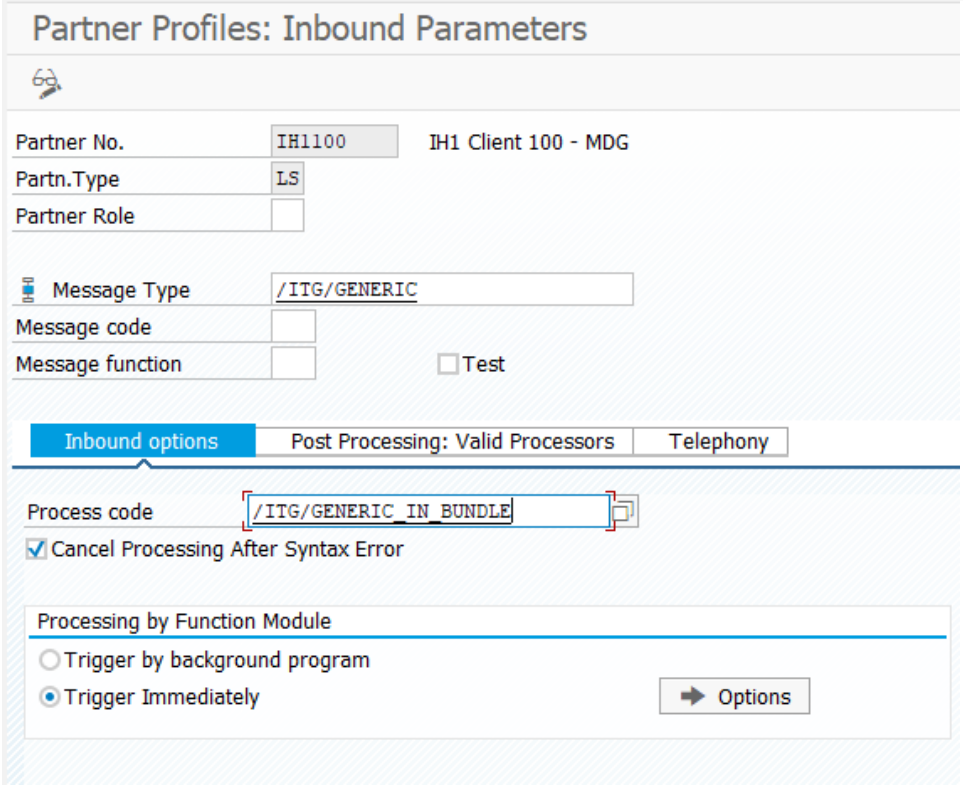

Display IMG
Existing BC Sets
BC Sets for Activity
Structure
IDoc Interface / Application Link Enabling (ALE)
Basic Settings
Communication
Modelling and Implementing Business Processes
Global Organizational Units
Maintain Distribution Model and Distribute Views
Configure Predefined ALE Business Processes


Distribution Model
Edit
Goto
Environment
System
Help
Generate Partner Profiles
Change Partner Profile
Display Distribution Model
Check Models
System View
Filter Model Displa
Distribution Model
Description/Technical Name
Model Views
ALE_SYNC
ALE_SYNC . No short text exists
FICADISP30
FICADISP30. No short text exists
CRM Scenarios
CRMSZ
Customizing Data Synchronization
CONTRLDATA
Example of MM contract distribution (filering at header level)
MM-PUR1
Example of MM contract distribution (filtering at item level)
MM-PUR2
Example of distributing test settings
QM-CONTR
HR <-> FI Scenario
HRFICOUPLI
Internet Scenarios
INTERNET
Logistics Scenarios
LOGISTICS
Master Data Distribution (MDM)
MASTERDATA
RDM Replication
RDM_REP
IH1100
IH1100 . No short text exists
IR1200
IR1200 . No short text exists
/ITG/GENERIC
RDM: generic message type

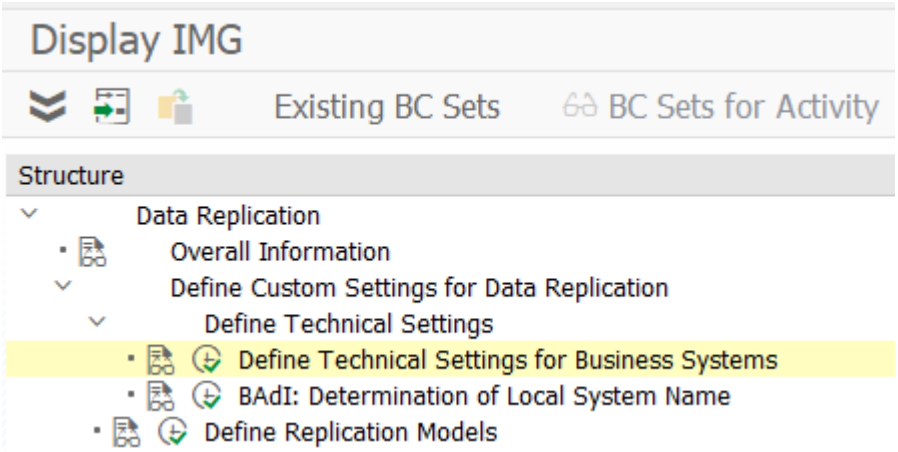


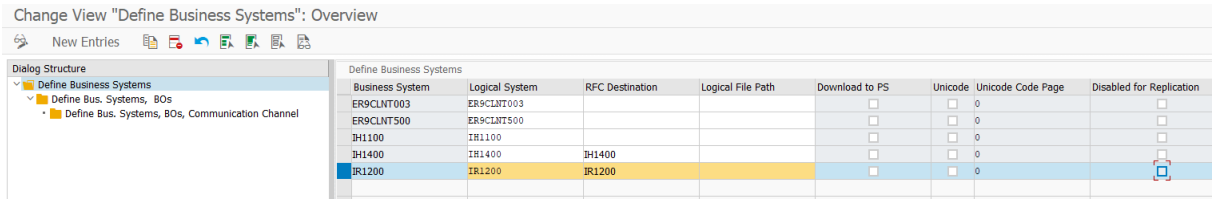
Check in WE20



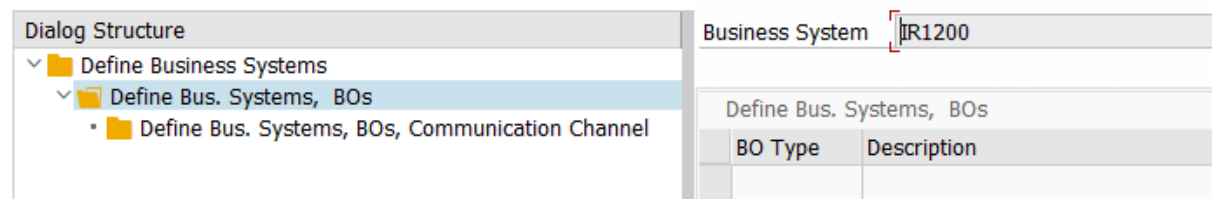


System IH1 100: Check DRFIMG:

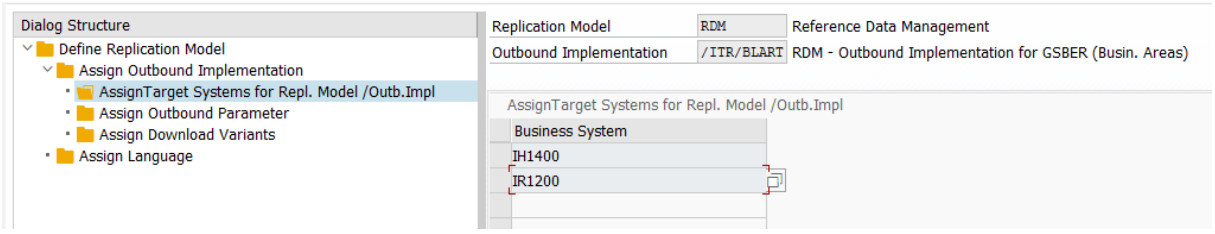


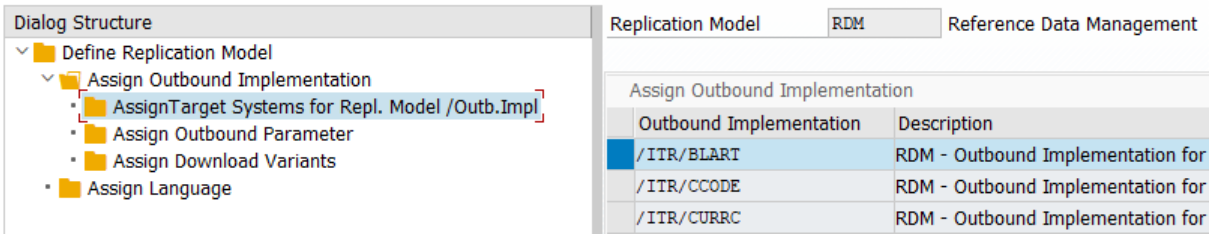


Nothing else to be configured:

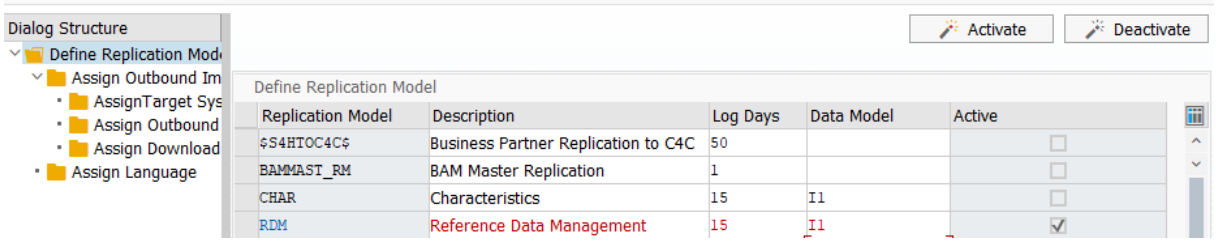


For every object in scope for this receiver system define target system and outbound implementation:





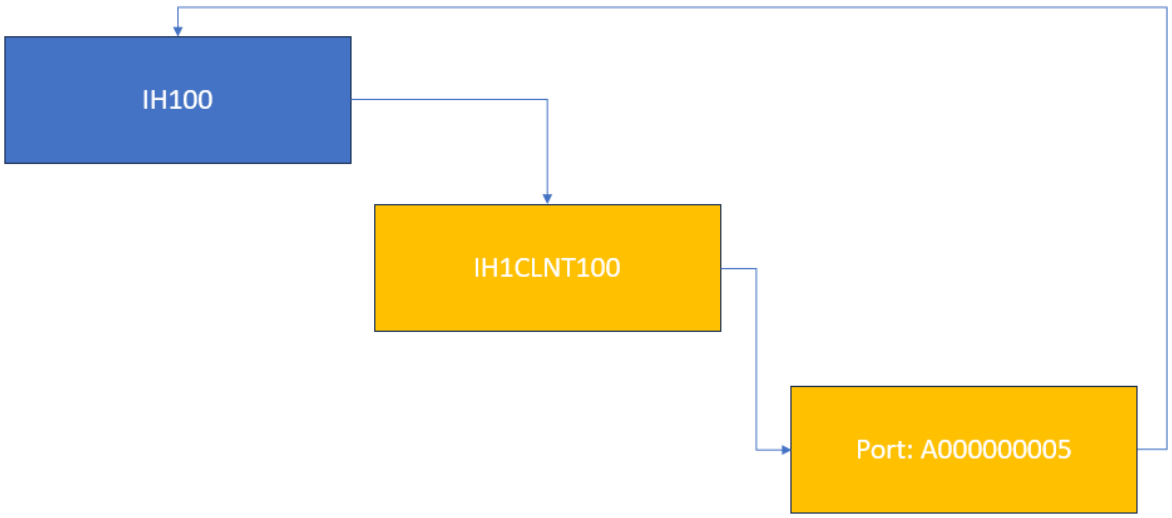
Data Replication Model needs to be active:



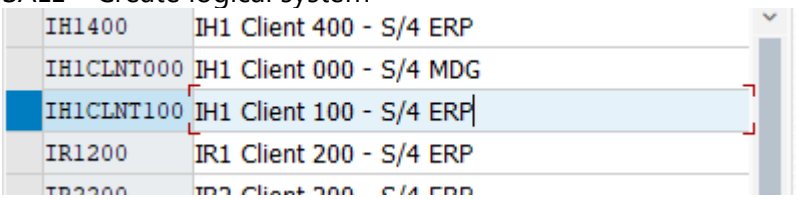
5.1.5 Business Governance: Data Replication to MDG client

This section shows a possible data replication configuration for the data replication from the MDG for RDM data persistence (based on the "Flex Mode") to the ERP tables within the same client. This is necessary to avoid inconsistencies for all object types that are in the scope for the Business Governance scenario. It is important to understand that (like in MDG for Financials) the data persistence is decoupled from the corresponding ERP tables. Without this replication scenario below the data between the ERP layer and the MDG layer will start to differ after the data has been loaded (from the ERP tables) and is changed for the first time in MDG for RDM.

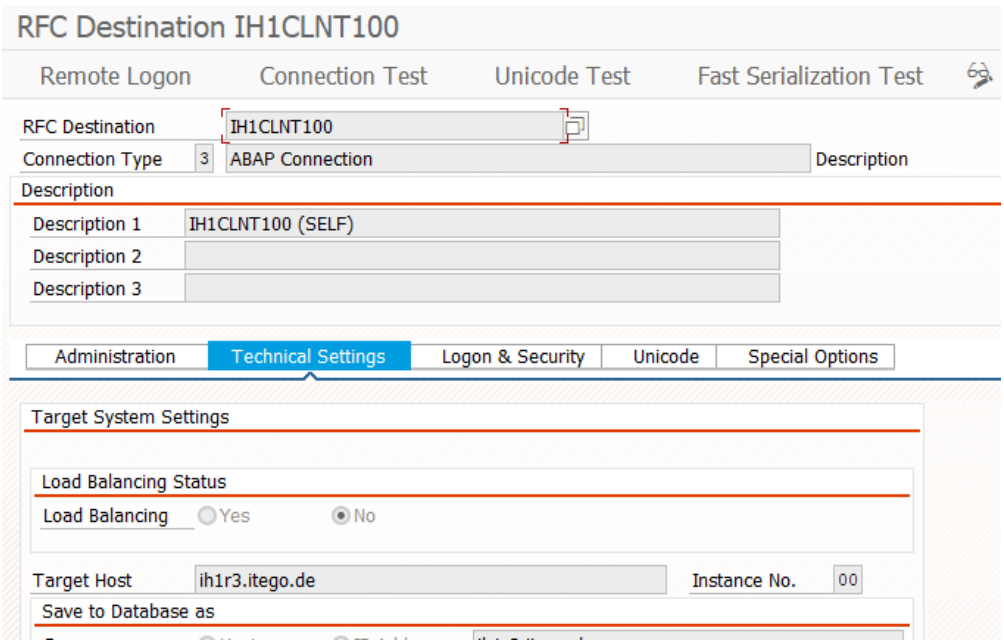
Create a new logical system as receiver: IH1CLNT100 and send back to IH1100

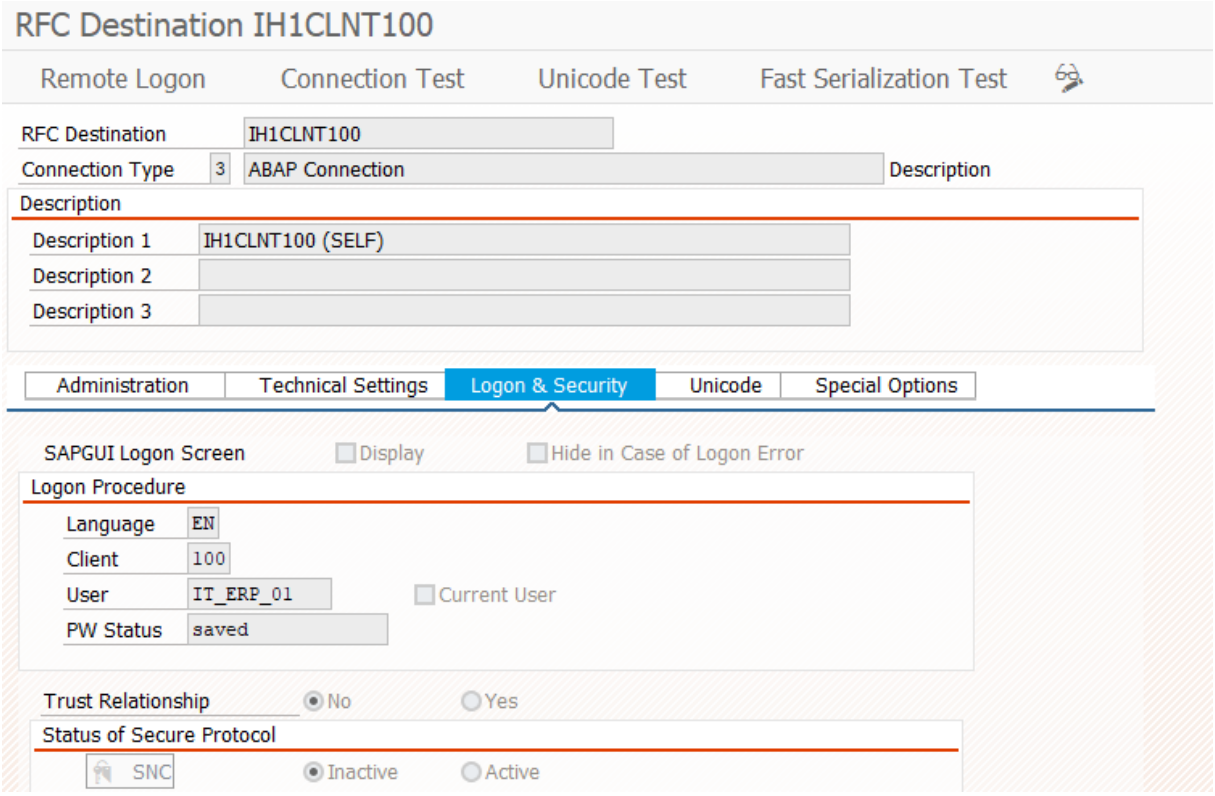


SALE – Create logical system



SM59: Create RFC Destination



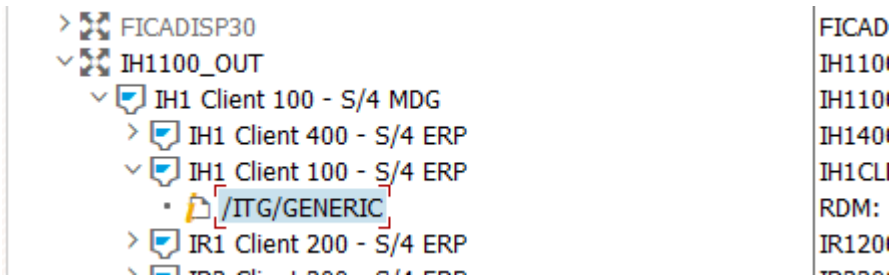


Note: A specific technical user needs to be defined and used. The screenshot only shows an example.

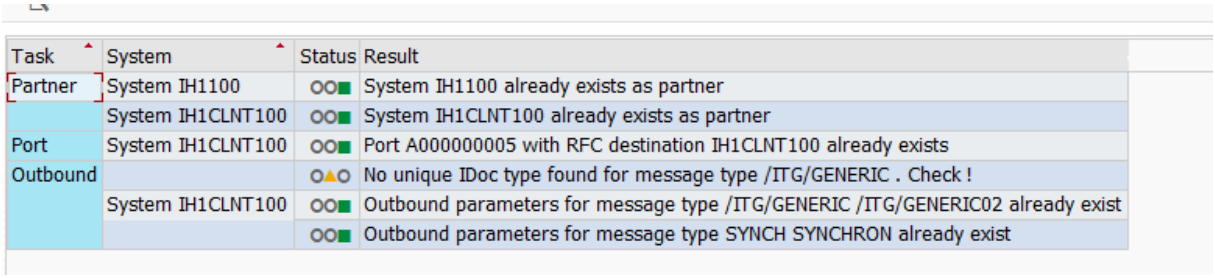
Connection Test

Action	Result
Logon	2 msec
Transfer of 0 KB	0 msec
Transfer of 10 KB	0 msec
Transfer of 20 KB	0 msec
Transfer of 30 KB	0 msec

Add to Distribution Model

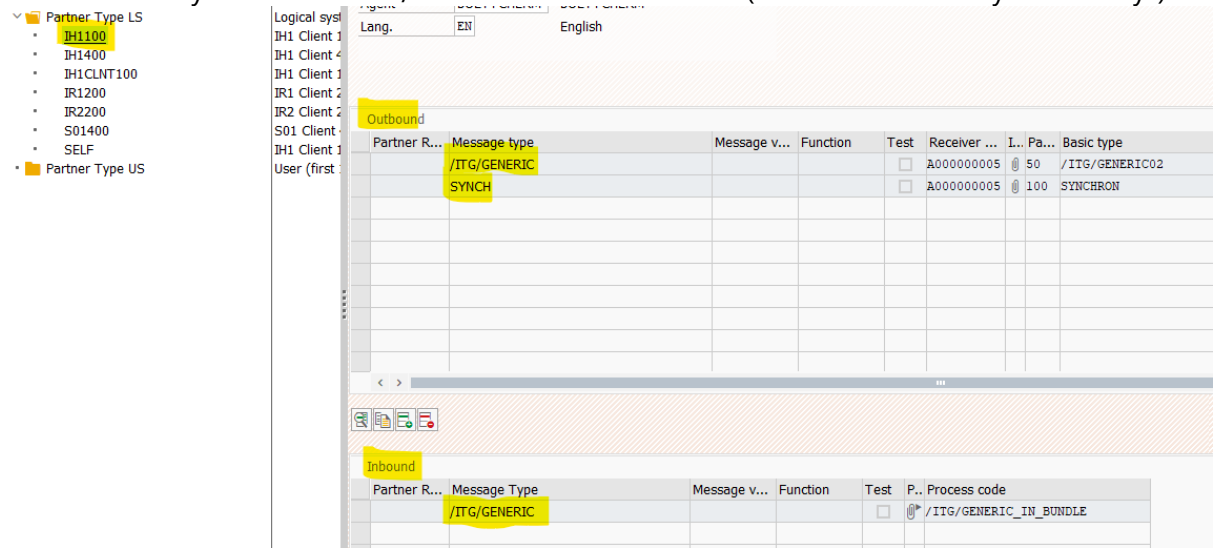


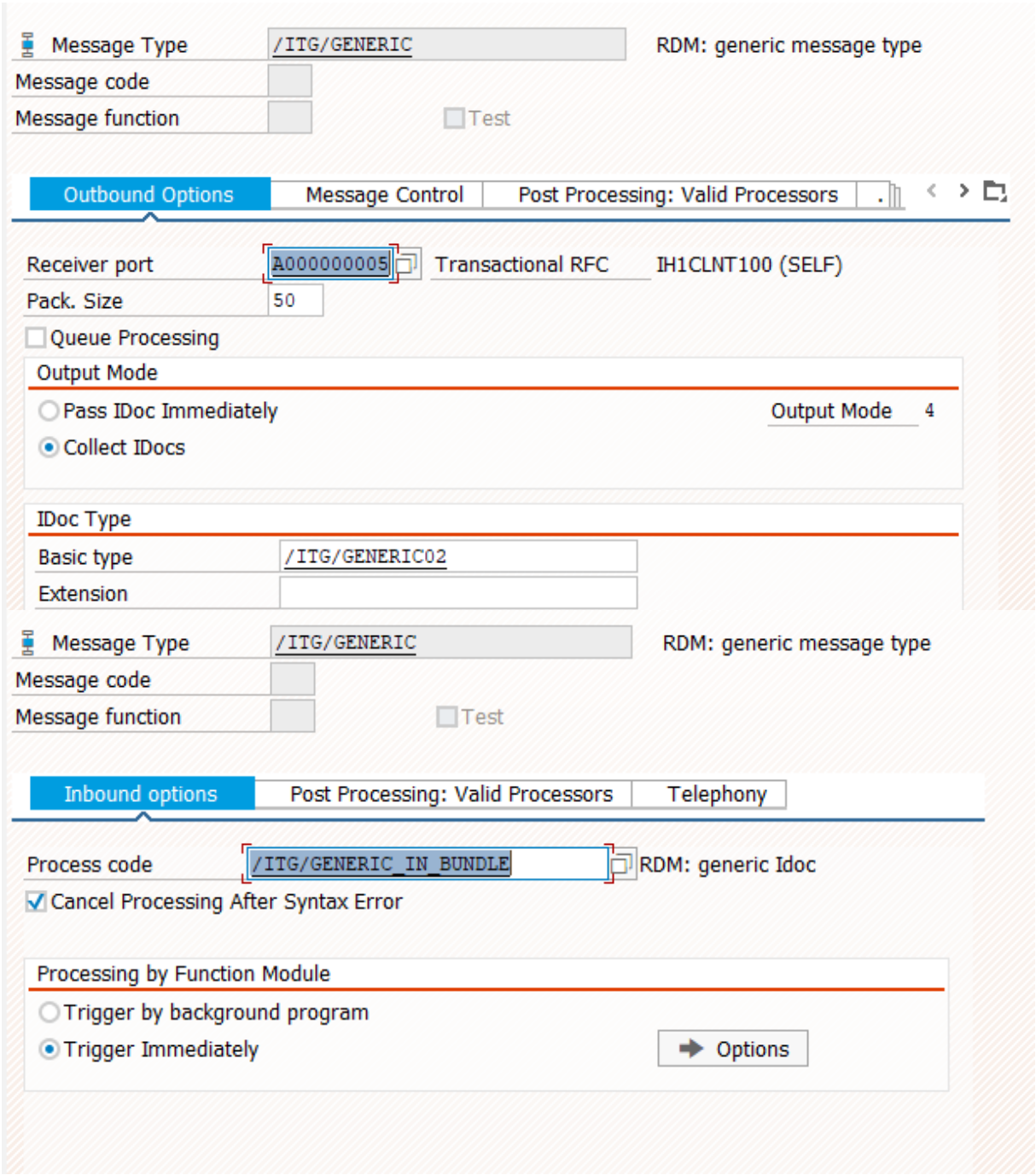
Generate Partner Profils



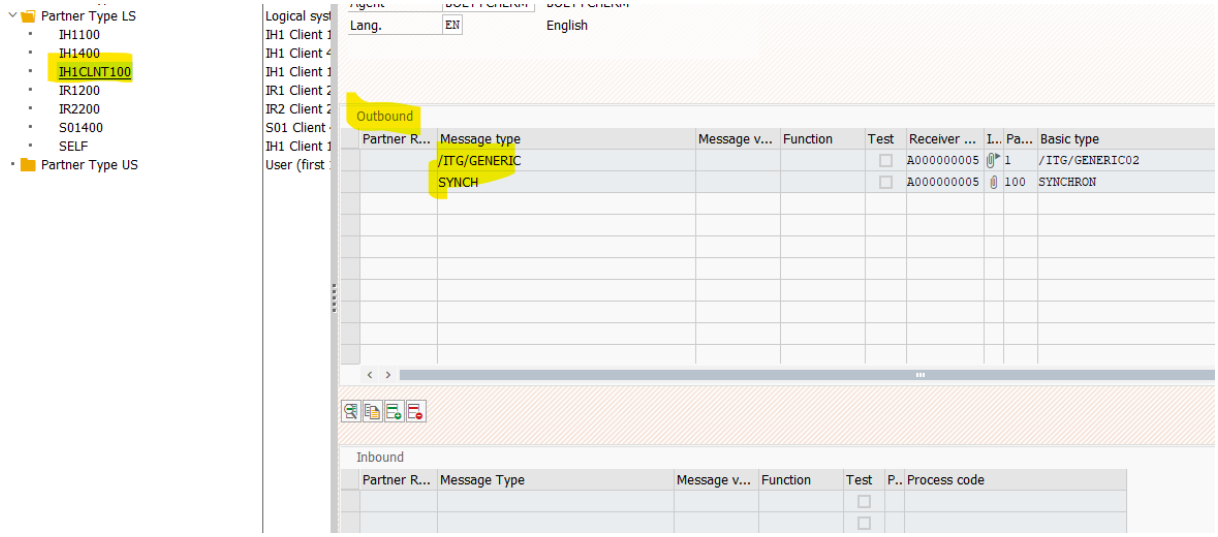
Task	System	Status	Result
Partner	System IH1100		System IH1100 already exists as partner
	System IH1CLNT100		System IH1CLNT100 already exists as partner
Port	System IH1CLNT100		Port A000000005 with RFC destination IH1CLNT100 already exists
Outbound			No unique IDoc type found for message type /ITG/GENERIC . Check !
	System IH1CLNT100		Outbound parameters for message type /ITG/GENERIC /ITG/GENERIC02 already exist
			Outbound parameters for message type SYNCH SYNCHRON already exist

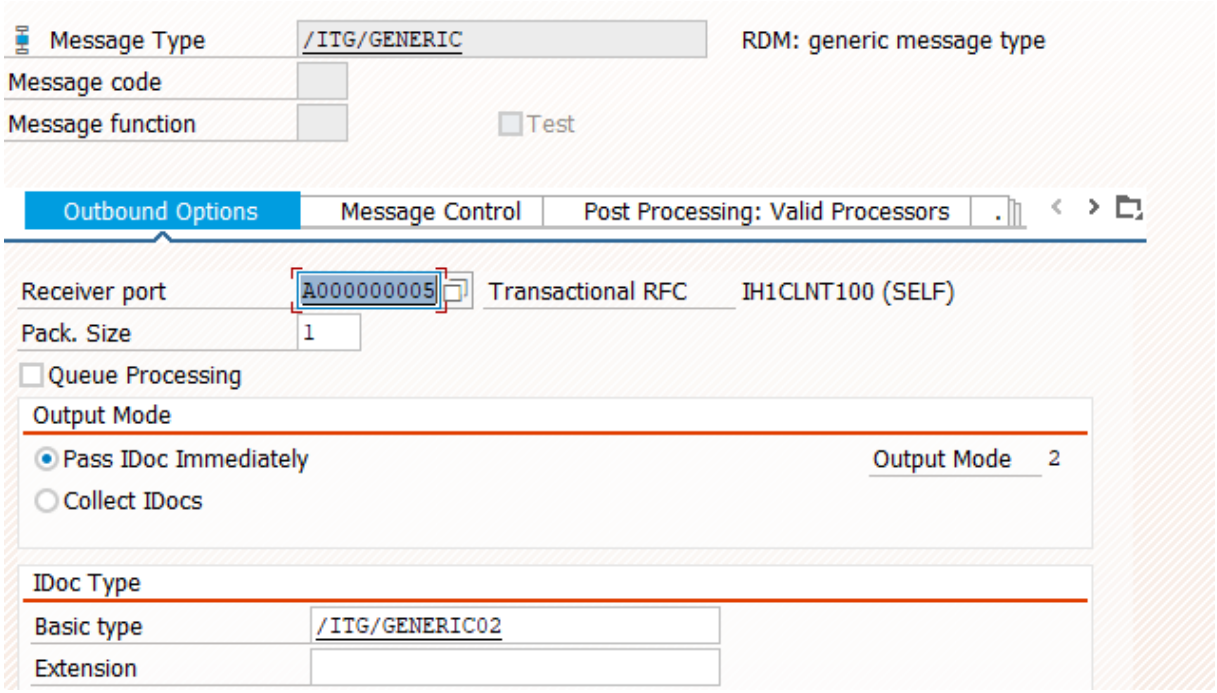
Check manually WE20: IH1100 / Outbound and Inbound! (IH1CLNT100 is only a “dummy”)



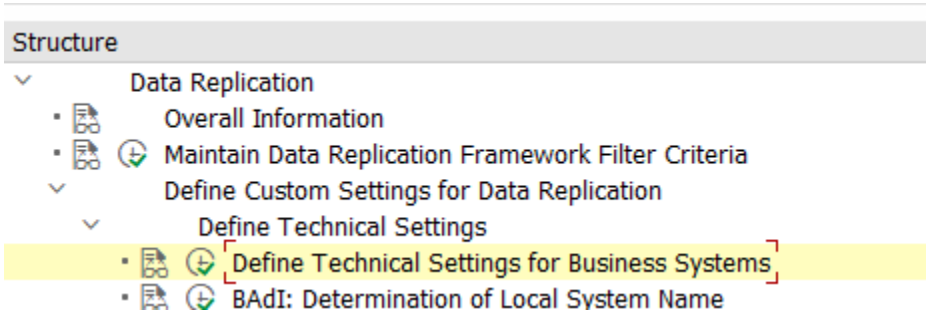


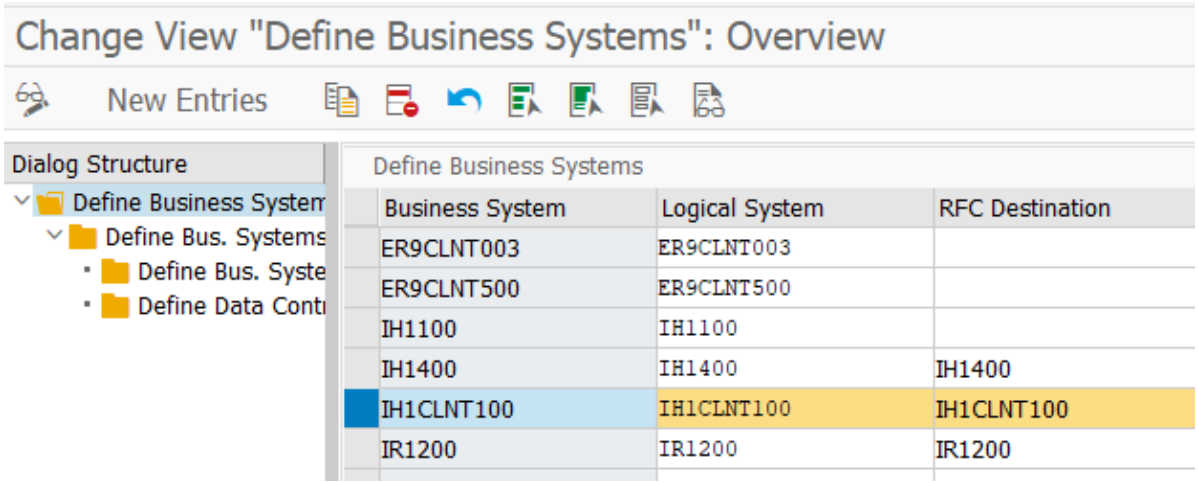
IH1CLNT100 / just Outbound:

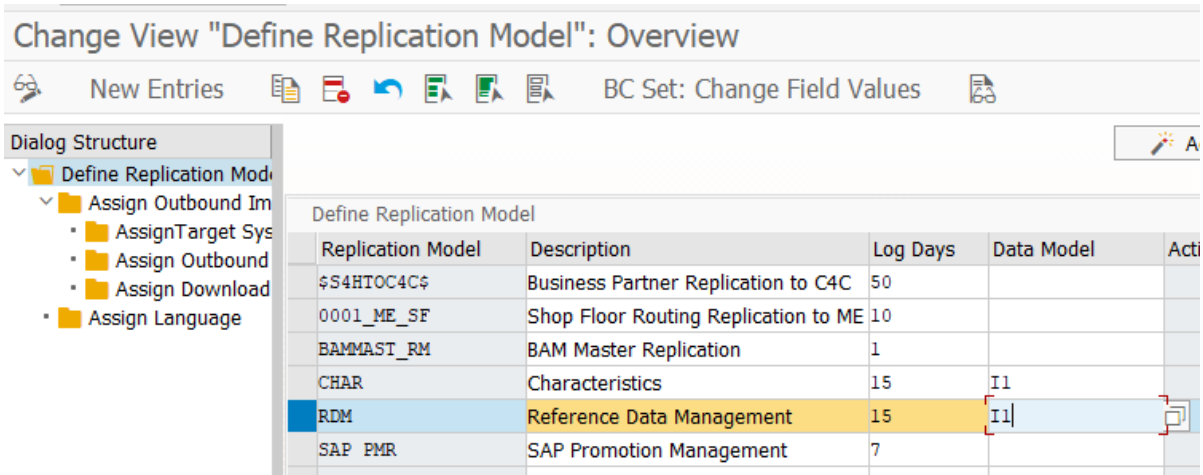

Partner Type LS
IH1100
IH1400
IH1CLNT100
IR1200
IR2200
S01400
SELF
Partner Type US
Logical syst
IH1 Client 1
IH1 Client 4
IH1 Client 1
IR1 Client 2
IR2 Client 2
S01 Client
IH1 Client 1
User (first
Lang.
EN
English
Outbound
Partner R...
Message type
Message v...
Function
Test
Receiver ...
I...
Pa...
Basic type
/ITG/GENERIC
SYNCH
A000000005
1
/ITG/GENERIC02
A000000005
100
SYNCHRON
Inbound
Partner R...
Message Type
Message v...
Function
Test
P..
Process code


Message Type
/ITG/GENERIC
RDM: generic message type
Message code
Message function
Test
Outbound Options
Message Control
Post Processing: Valid Processors
Receiver port
A000000005
Transactional RFC
IH1CLNT100 (SELF)
Pack. Size
1
Queue Processing
Output Mode
Pass IDoc Immediately
Output Mode
2
Collect IDocs
IDoc Type
Basic type
/ITG/GENERIC02
Extension

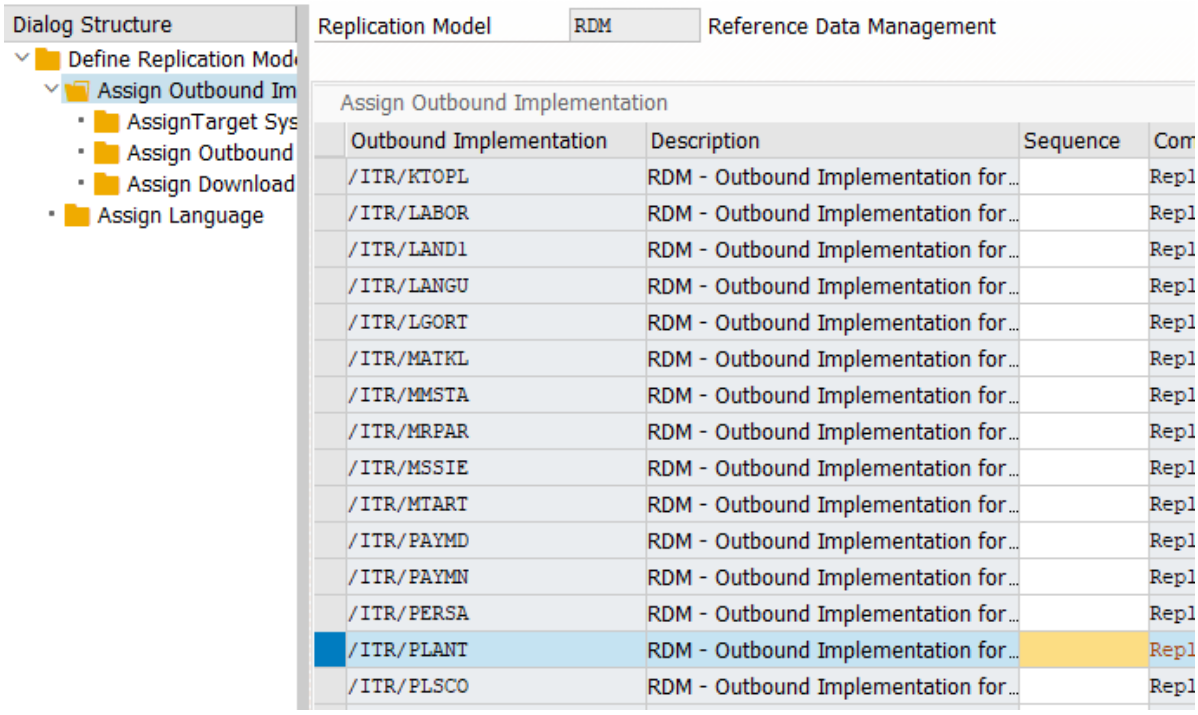
DRFIMG:

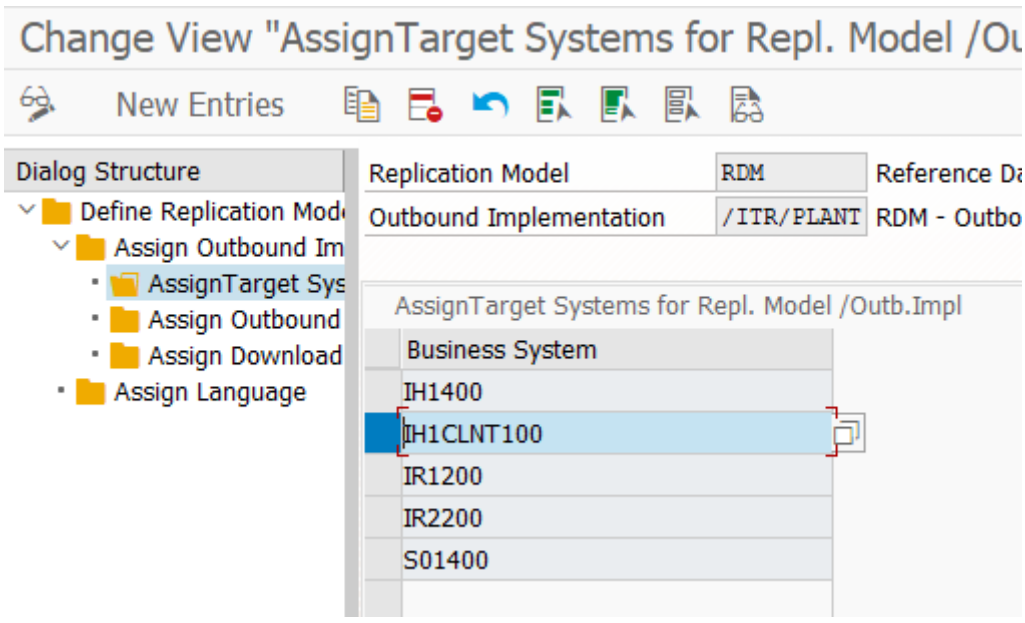




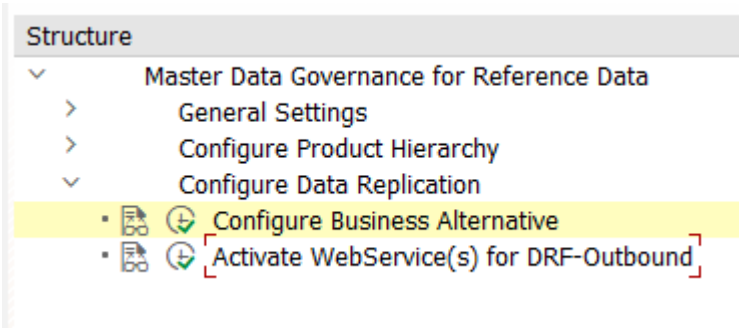


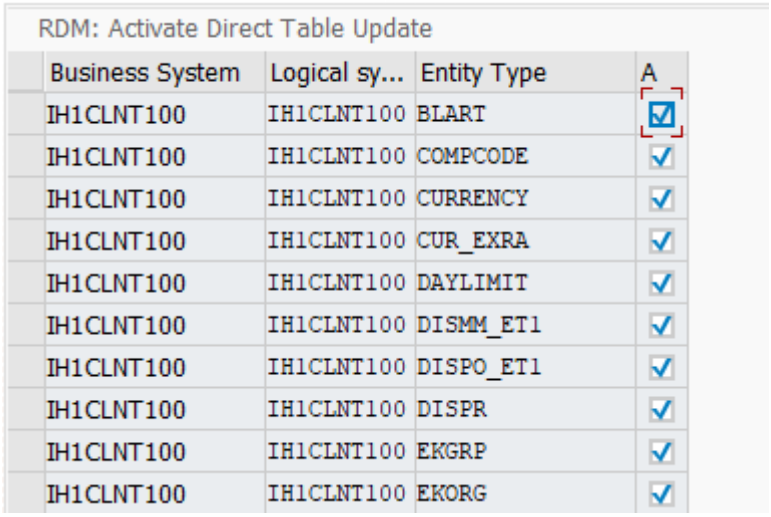
The next step needs to be done for each object type in scope for Business Governance ("Direct Update"). The below list is just an example.





Last step: Activate the Business Governance for all object types in scope for Business Governance ("Direct Update"). The below list is again just an example.





5.2 Data Replication based on Webservices

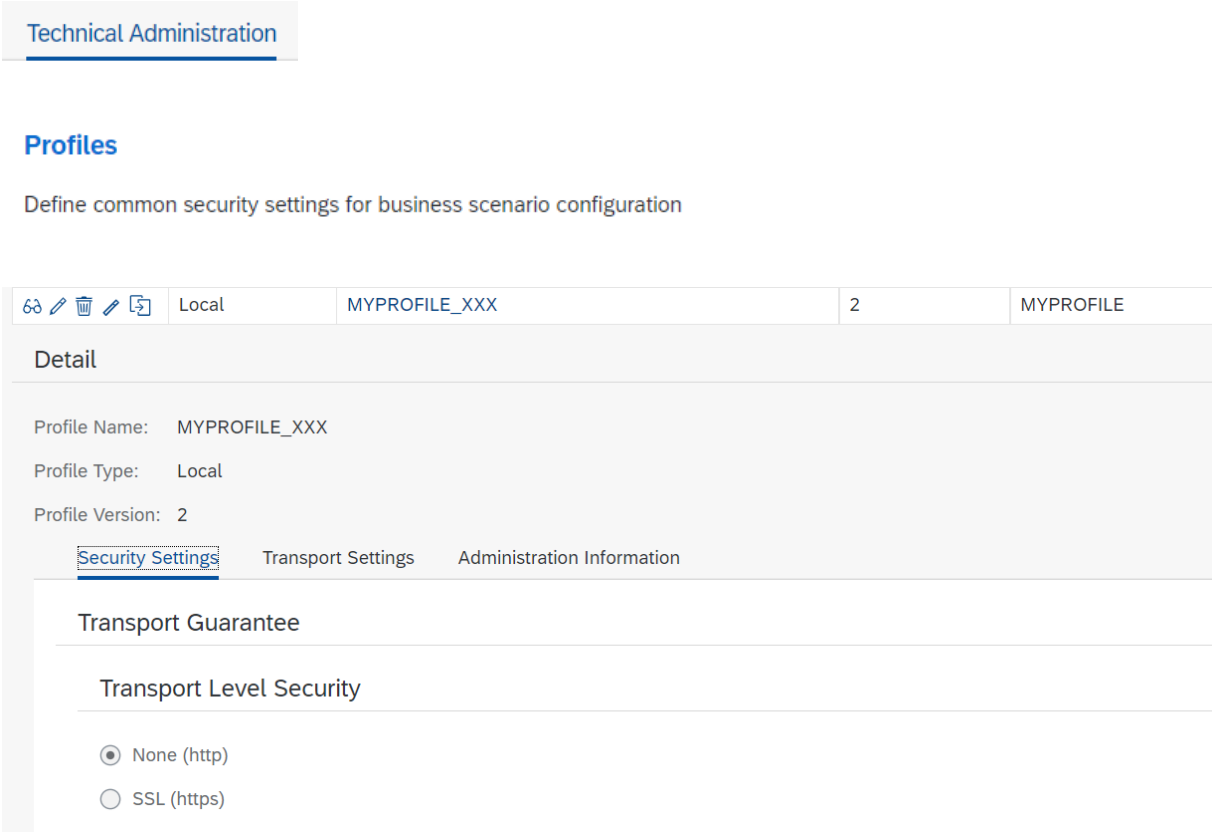
This section shows how to set up a Webservice based communication using the following systems:

- RDM for MDG (sender): System IH1, client 100
- SAP Business System (receiver): System IR1, client 200

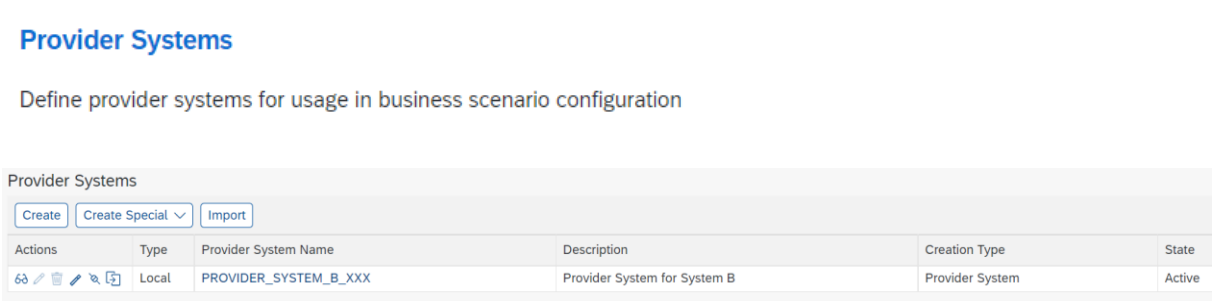
Please adjust the examples below to your own system landscape.

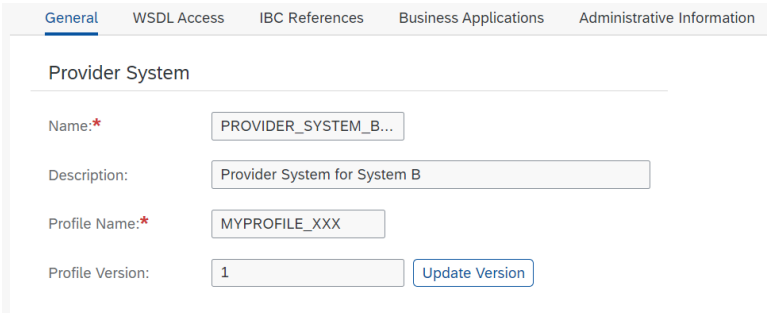
5.2.1 Receiver: Technical Administration: Profiles and Provider Systems

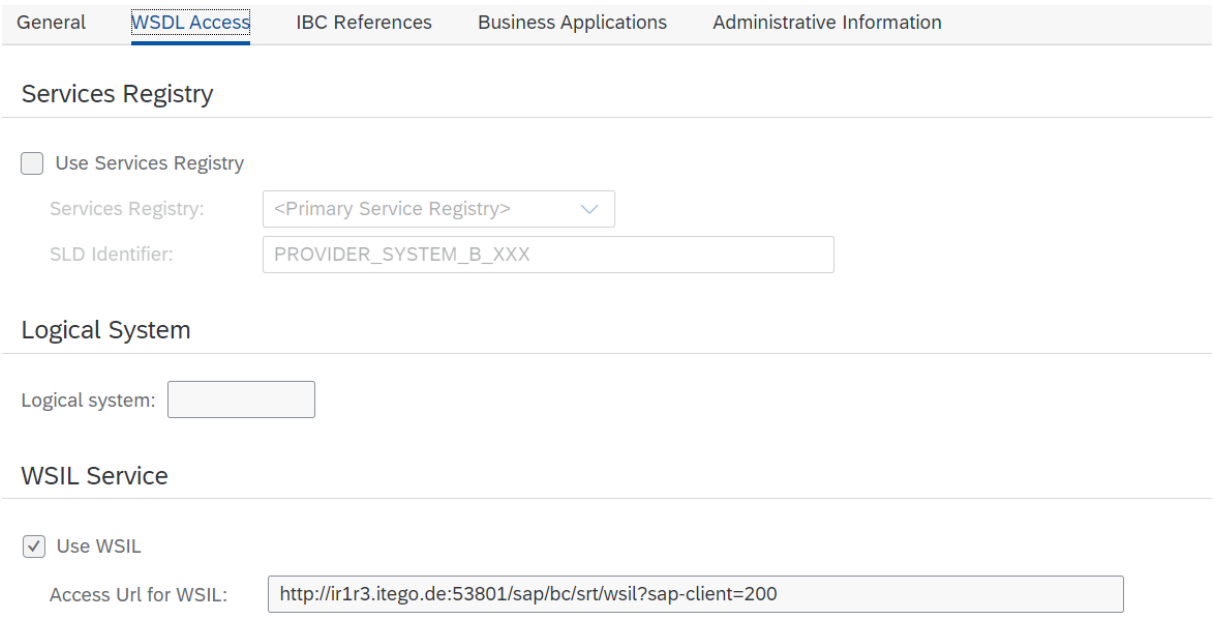
System IR1 200: transaction SOAMANAGER:

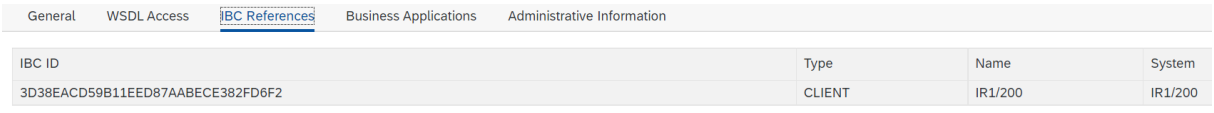


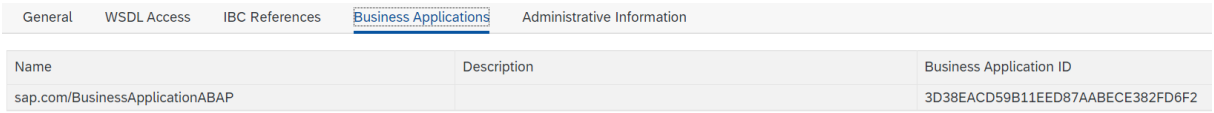
Or https based on your security requirements.





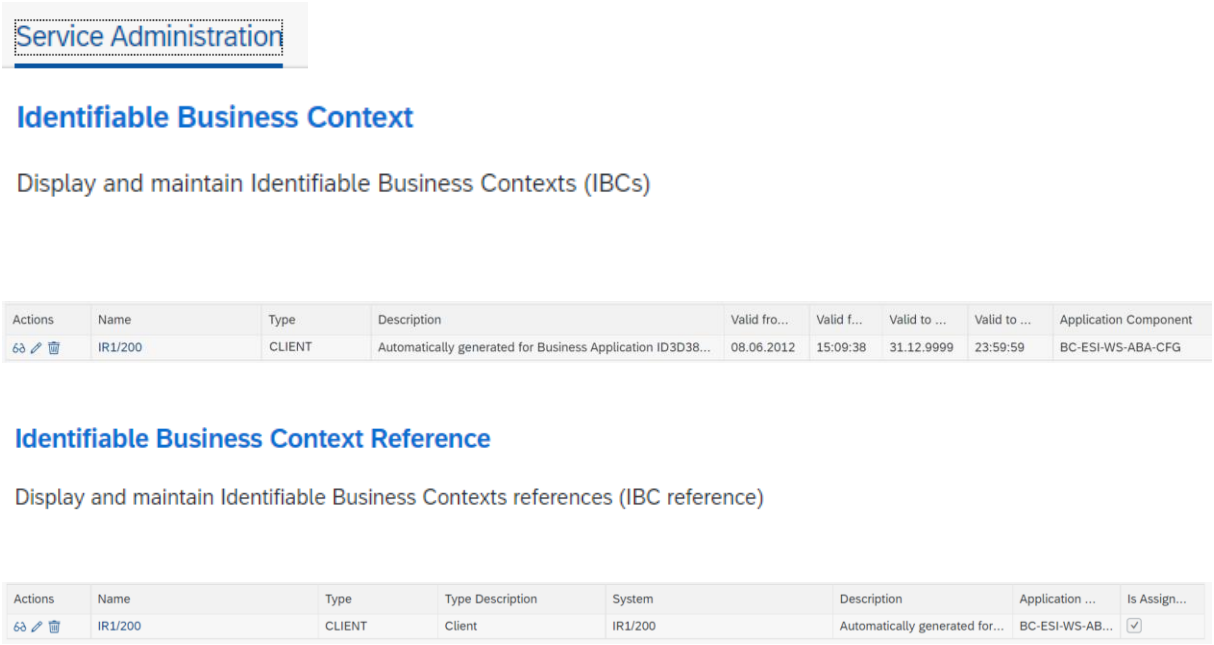






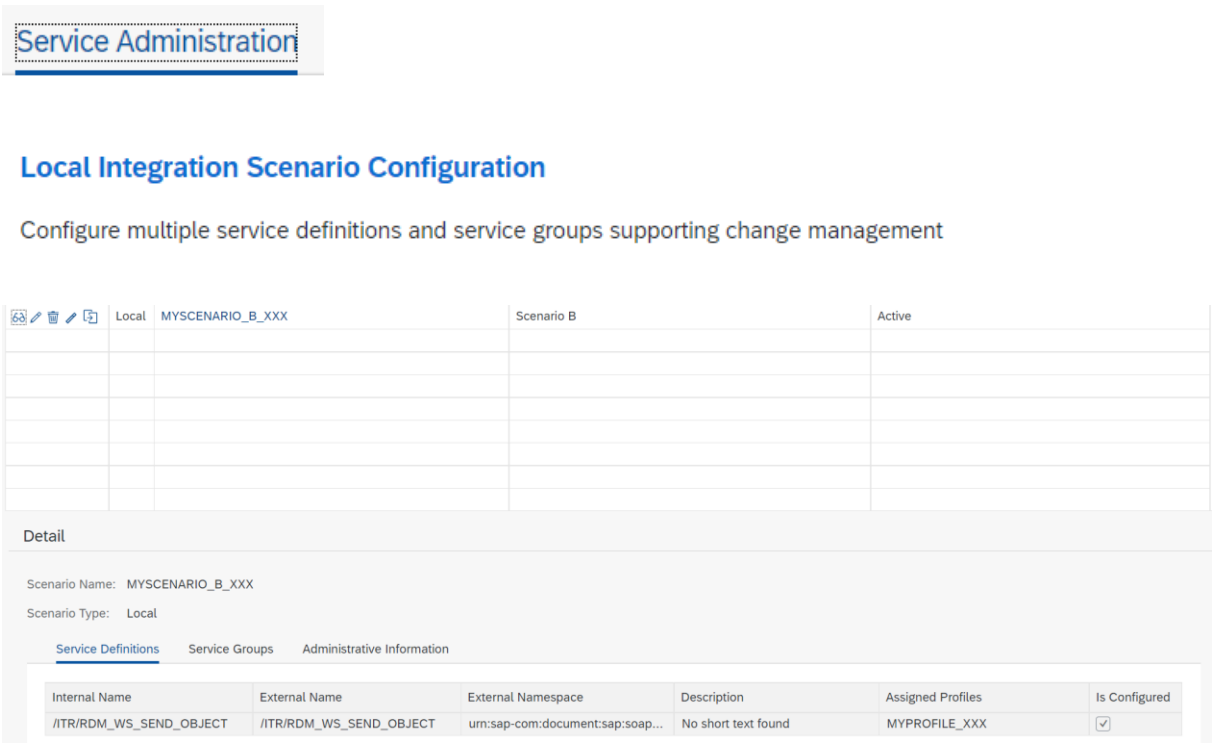
5.2.2 Receiver: Service Administration: Business Context

System IR1 200: transaction SOAMANAGER:



5.2.3 Receiver: Service Administration: Local Integration Scenario

System IR1 200: transaction SOAMANAGER:



5.2.4 Receiver: Services Registry: Published Systems and Objects

System IR1 200: transaction SOAMANAGER:

Services Registry

Publishing Systems

Display and maintain Publishing Systems in Services Registry

Actions	Name	Name and Client	Logical Key	Host Name	Publishing System Type	Application	Publ.	Orig.
	IR1	IR1(200)	200.SystemName.IR1.SystemNumber.0090257000.SystemHome.ir1r3#ABAP	ir1r3	ABAP	☑	☑	☑

Published Service Definitions

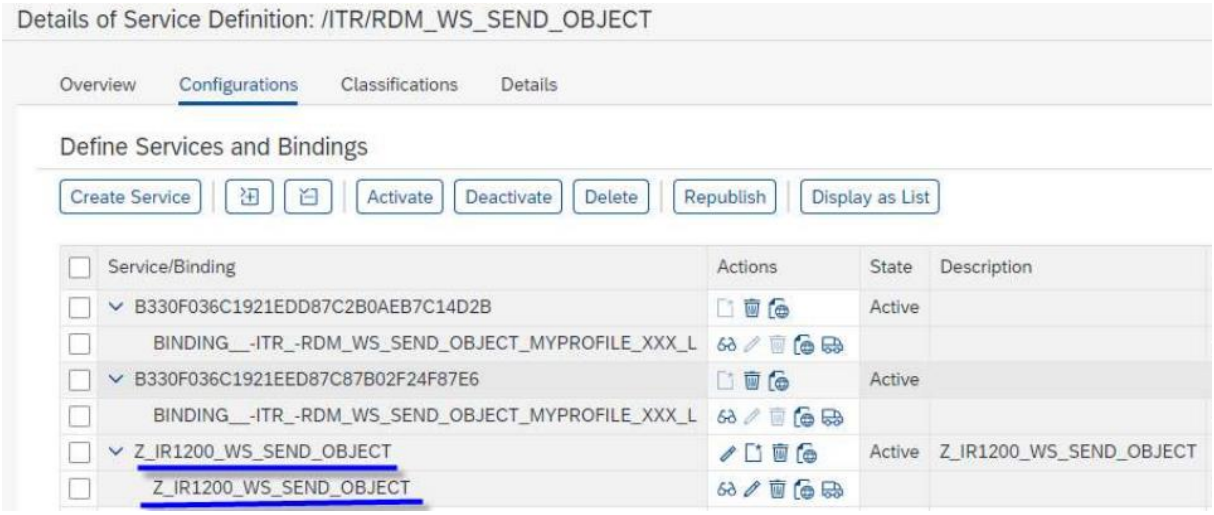
Display and maintain published Service Definitions in Services Registry

Actions	Internal Name	External Namespace	External Name	State	Description	Publishing System
	/ITR/RDM_WS_SEND_OBJECT	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_SEND_OBJECT	Configured	No short text found	IR1(200) on ir1r3

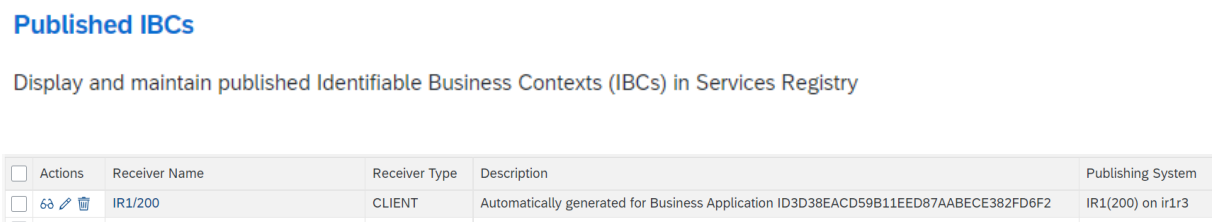
Published Bindings

Display and maintain published Bindings in Services Registry

Actions	Binding Name	Service Namespace	Internal Service Name	External Service Name	Publishing Sys
	BINDING__-ITR_-RDM_WS_SEND_OBJECT_MYPROFILE_XXX_L	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_SEND_OBJECT	/ITR/RDM_WS_SEND_OBJECT	IR1(200) on ir:
	BINDING__-ITR_-RDM_WS_SEND_OBJECT_MYPROFILE_XXX_L	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_SEND_OBJECT	/ITR/RDM_WS_SEND_OBJECT	IR1(200) on ir:
	Z_IR1200_WS_SEND_OBJECT	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_SEND_OBJECT	/ITR/RDM_WS_SEND_OBJECT	IR1(200) on ir:

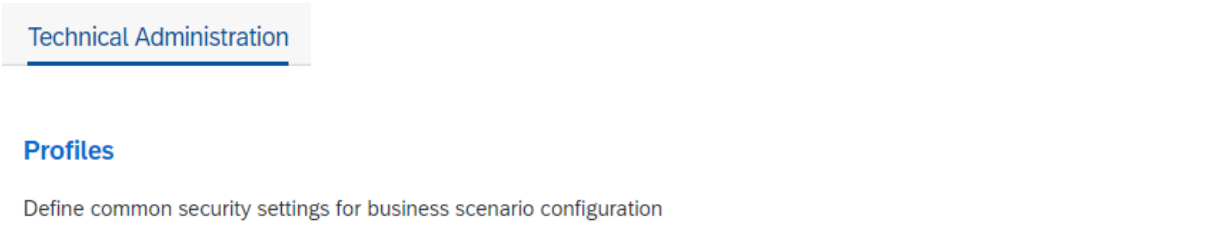


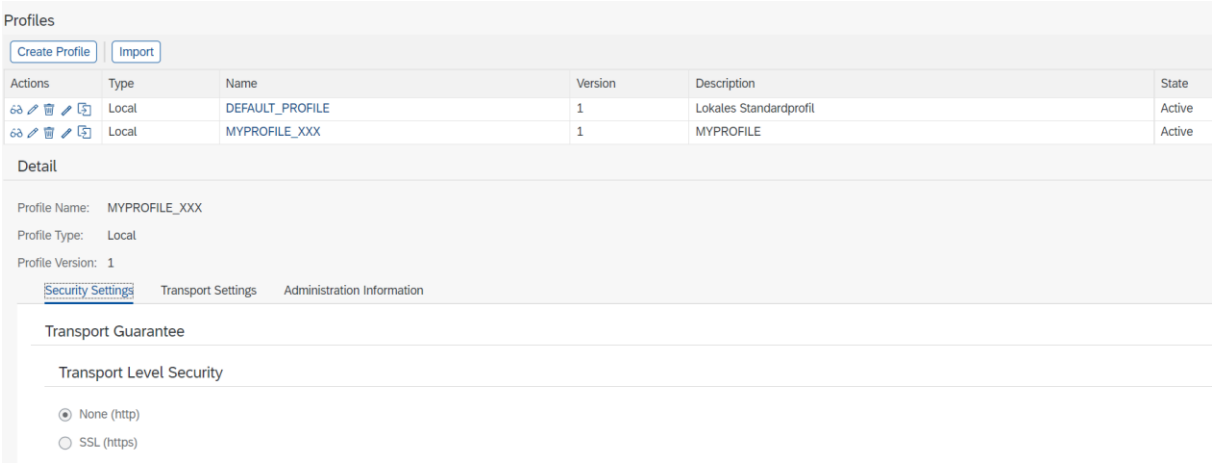
Export WSDL to set up sender system in a later step.



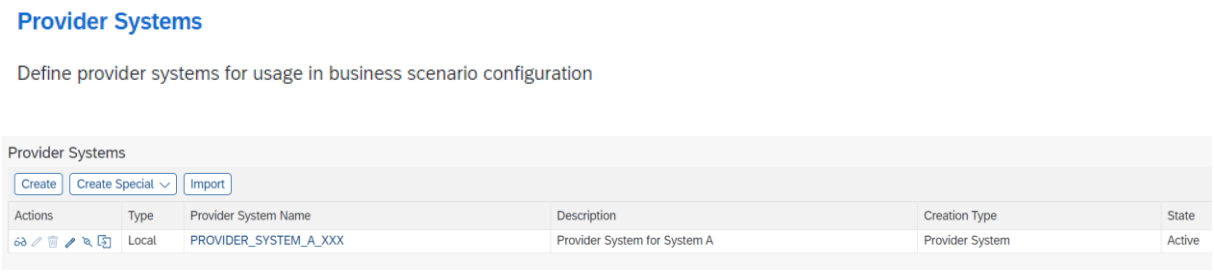
5.2.5 Sender: Technical Administration: Profiles and Provider Systems

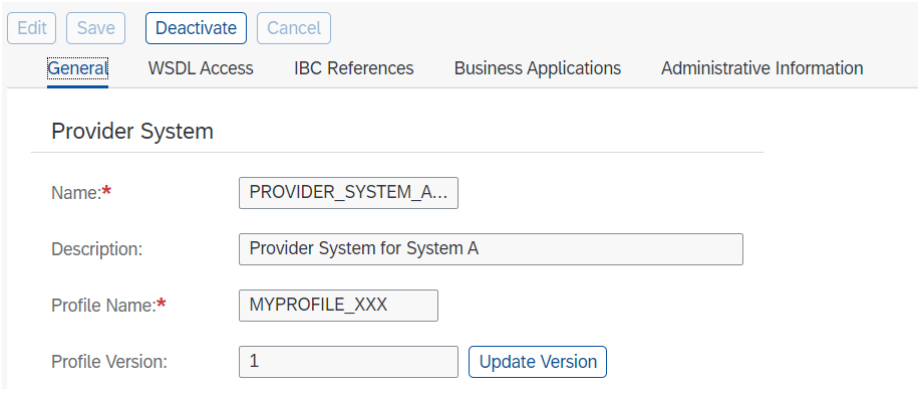
System IH1 100: transaction SOAMANAGER:

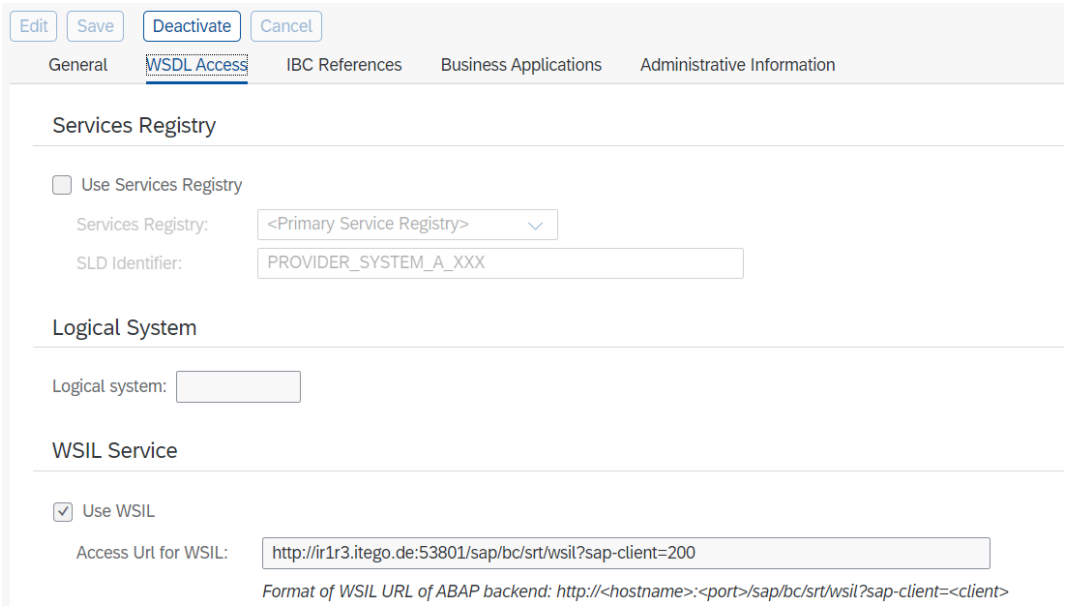


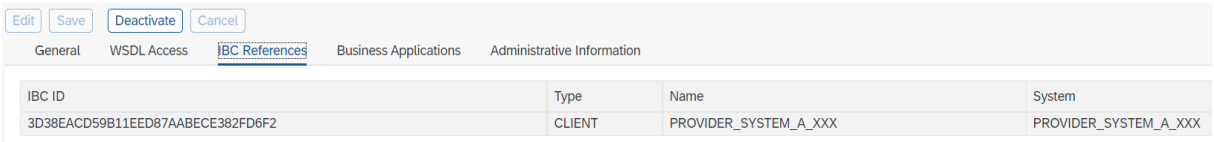


Or https based on your security requirements.



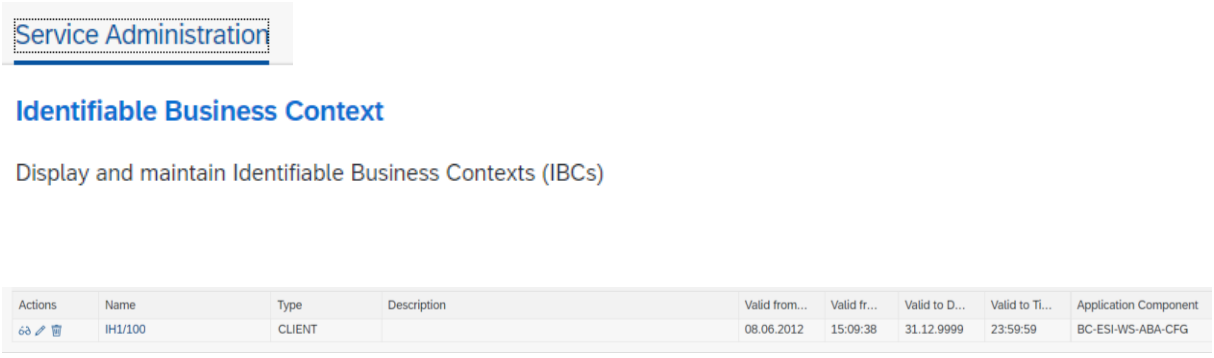






5.2.6 Sender: Service Administration: Business Context

System IH1 100: transaction SOAMANAGER:



Identifiable Business Context Reference

Display and maintain Identifiable Business Contexts references (IBC reference)

Actions	Name	Type	Type Description	System	Description	Application Co...	Is Assigned
	IH1/100	CLIENT	Client	IH1/100		BC-ESI-WS-ABA...	☐
	PROVIDER_SYSTEM_A_XXX	CLIENT		PROVIDER_SYSTEM_A_XXX	Automatically generated for Bu...	BC-ESI-WS-ABA...	☑

5.2.7 Sender: Service Administration: Local Integration Scenario

System IH1 100: transaction SOAMANAGER:

Service Administration

Local Integration Scenario Configuration

Configure multiple service definitions and service groups supporting change management

Local	MYSCENARIO_A_XXX	Consumer Scenario	Active

Detail

Scenario Name: MYSCENARIO_A_XXX

Scenario Type: Local

Service Definitions | Service Groups | Administrative Information

Internal Name	Description	Provider IBC Reference	Communication Protocol	Is Configured
/ITR/RDM_WS_SEND		CLIENT / PROVIDER_SYST...	Use WS Protocol	☑

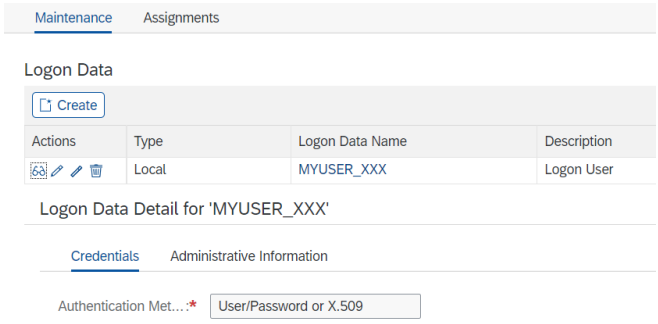
5.2.8 Sender: Service Administration: Logon Data

System IH1 100: transaction SOAMANAGER:

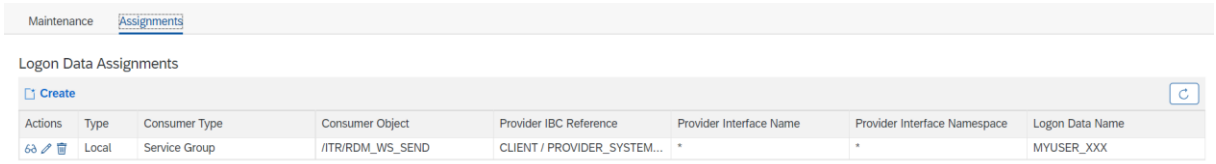
Service Administration

Logon Data Management

Define logon data used by business scenario configuration



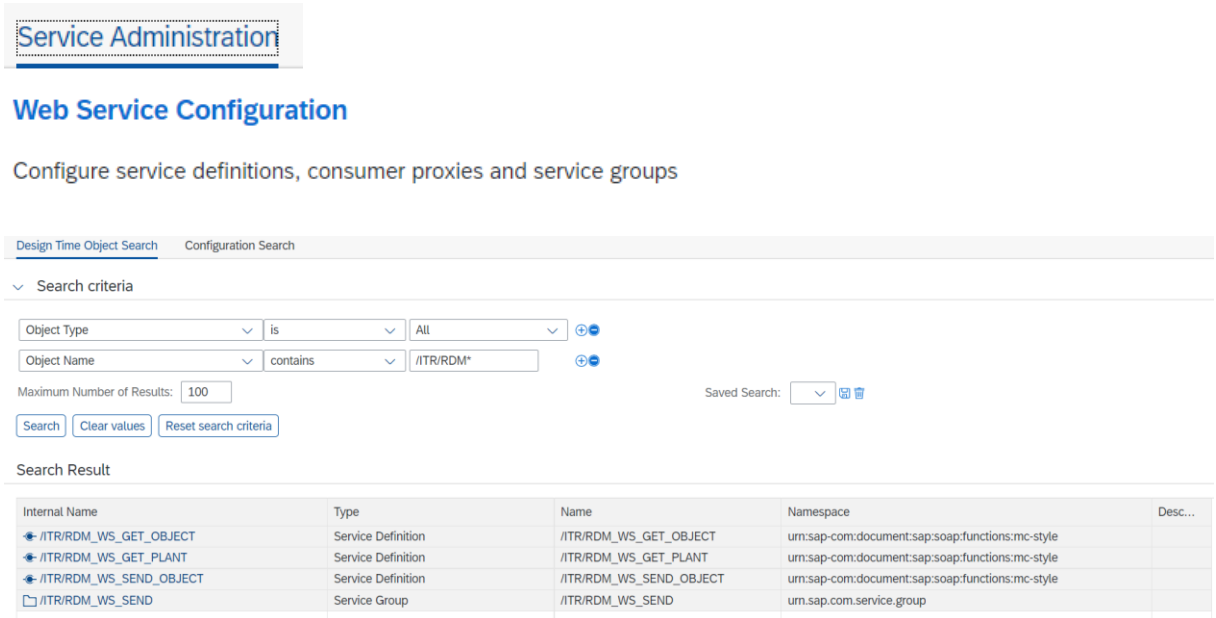
Provide User and Password



Assignment to Service Group

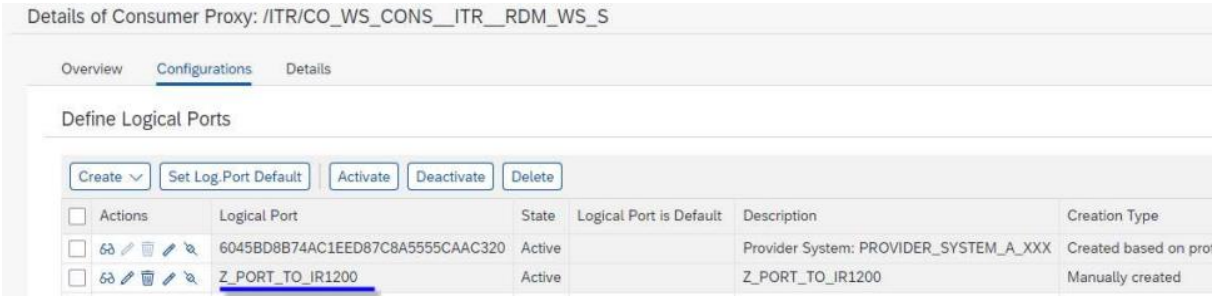
5.2.9 Sender: Service Administration: Web Service

System IH1 100: transaction SOAMANAGER:



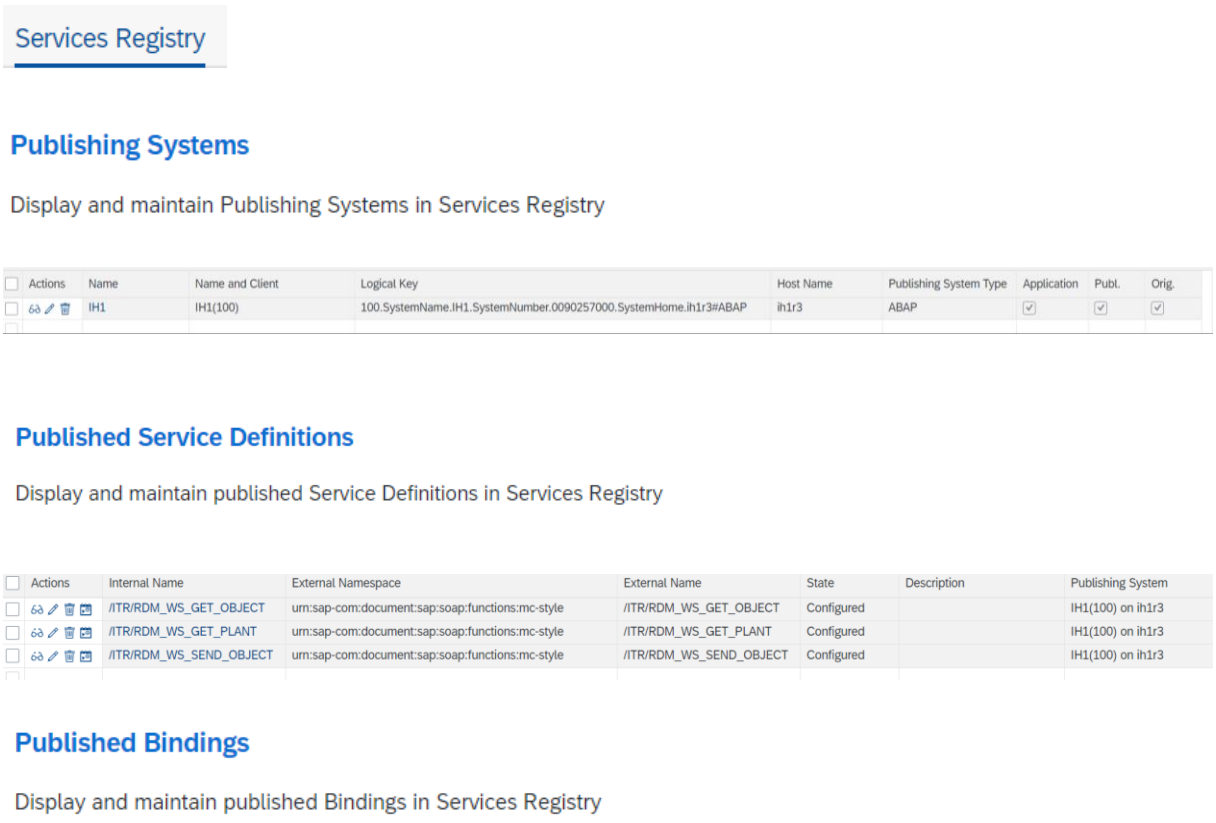
Internal Name	Actions	Binding/Log.Port	Type	State	Creation Type
/ITR/CO_WS_CONS__ITR__RDM_WS_S		6045BD8B74AC1EED87C8A5555CAAC320	Logical Port	Active	Created based on profile MYPROFILE_XXX/1/Local
		Z_PORT_TO_IR1200	Logical Port	Active	Manually created
/ITR/RDM_WS_GET_OBJECT		Z_RDM_GET_OBJECT	Binding	Active	Manually created
/ITR/RDM_WS_GET_PLANT		Z_RDM_GET_PLANT	Binding	Active	Manually created
/ITR/RDM_WS_SEND_OBJECT		Z_RDM_SEND_OBJECT	Binding	Active	Manually created

Logical port created based on exported WSDL from receiver system.



5.2.10 Sender: Services Registry: Published Systems and Objects

System IH1 100: transaction SOAMANAGER:



Actions	Binding Name	Service Namespace	Internal Service Name	External Service Name	Publishing System
	Z_RDM_GET_OBJECT	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_GET_OBJECT	/ITR/RDM_WS_GET_OBJECT	IH1(100) on ih1r3
	Z_RDM_GET_PLANT	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_GET_PLANT	/ITR/RDM_WS_GET_PLANT	IH1(100) on ih1r3
	Z_RDM_SEND_OBJECT	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_SEND_OBJECT	/ITR/RDM_WS_SEND_OBJECT	IH1(100) on ih1r3

Published IBCs

Display and maintain published Identifiable Business Contexts (IBCs) in Services Registry

Actions	Receiver Name	Receiver Type	Description	Publishing System
	IH1/100	CLIENT		IH1(100) on ih1r3