Reference Data Management 2307

for SAP Master Data Governance

Configuration

Version: 28.07.2023

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

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").

3.2 Activate Data Model I1

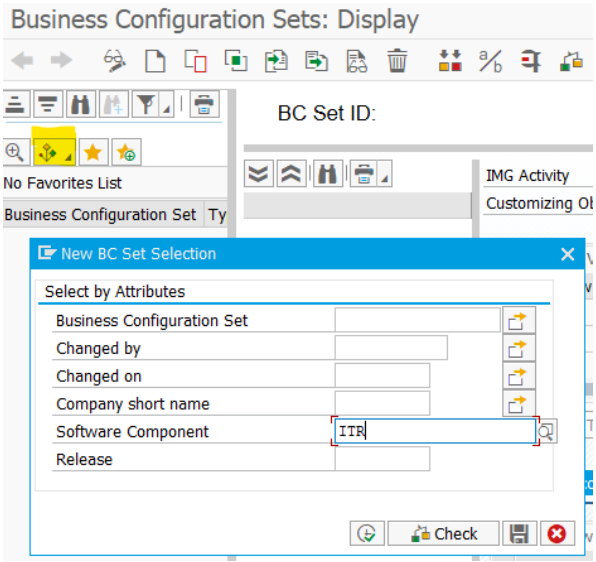
Check whether the data model I1 has been activated in transaction MDGIMG: General Settings -> Data Modeling -> Edit Data Model. If it has not been activated, select Data Model "I1" and click on  "Activate".

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).

3.3.1 SAP MDG: BC Sets

Note: If you choose to activate a BC Set, please activate all versions that are available starting with version <n>=01 for

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

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: These BC Sets deliver 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 these BC Sets.

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 3.9 Set Up Data Replication for additional information.

3.3.1.2 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

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 Sets have 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:

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

3.3.2.2 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 SAP Business system as a sender system for the synchronization or the consolidation of reference data):

- /ITR/RDH_CONFIG_GROUPS_<n> RDH Configuration Groups

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 3.8 "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_STAGING_MENU Local Staging Area NWBC Menu
- /ITR/MDG_STAGING_<n> Local Staging Area
- /ITR/MDG_LSA_ADMIN_<n> Local Staging Area Administration
- /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

3.4.3 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 Send-System
- /ITR/ITEGO_MDG_RDH_BUSINES_<n>S Business Functions in the Send-System
- /ITR/ITEGO_MDG_RDH_EXPERT_<n>S Expert Functions in the Send-System

3.4.4 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

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 from to your MDG Client, you need to execute the following steps:

1. **Create Transport**: Log into 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 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.

3.6.1 SAP MDG: Adjust Global 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.

3.6.2 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 please maintain the individual scope in view /ITR/LSA_TABS_FV via transaction SM30. Every attribute which is not within this scope definition will not be overwritten when data is activated on the business system.

3.7 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 this.

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

3.8 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**):

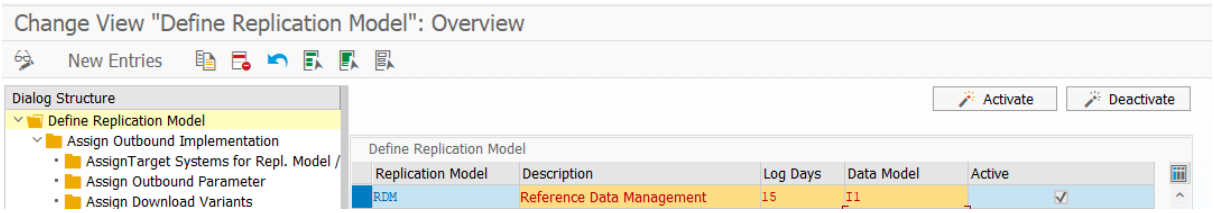
- /ITR/MDG_RDM_CREQUEST_<n> MDG RDM Change Request Types
- /ITR/MDG_RDM_WORKFLOW_<n> MDG RDM Workflow

3.9 Set Up Data Replication

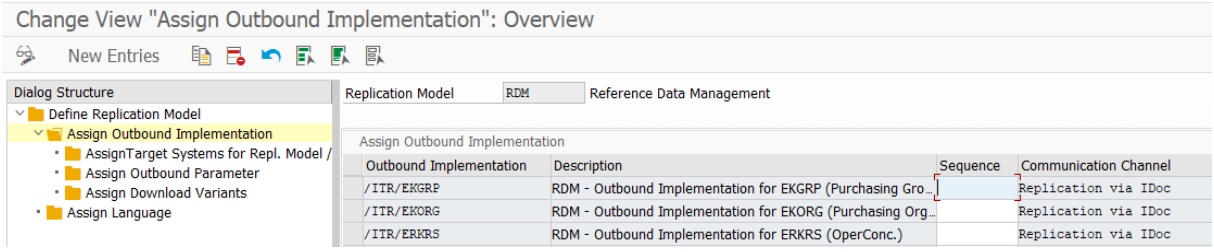
In order to configure the Data Replication for RDM the “Data Replication Framework” (DRF) needs to be configured as well as the technical settings for ALE or Webservices.

3.9.1 Data Replication Framework settings

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

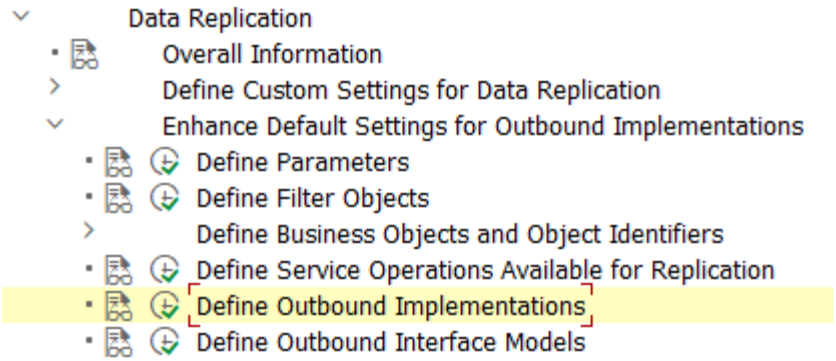


Configure Outbound Implementations for your model based on the scope of your RDM implementation and define a communication channel.



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

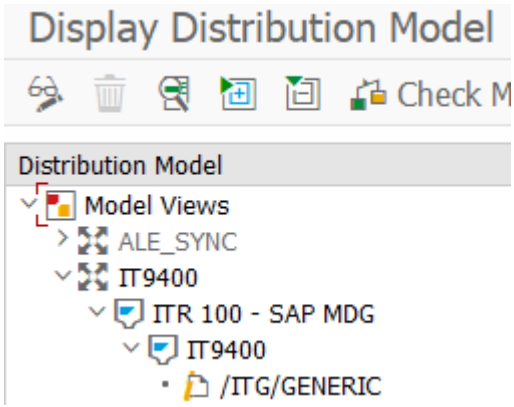
Note: You do not have to change the outbound implementation classes for other objects. They follow a generic approach even if the classes do not have a "_GEN" suffix.

3.9.2 Data Replication using Application Link Enabling (ALE)

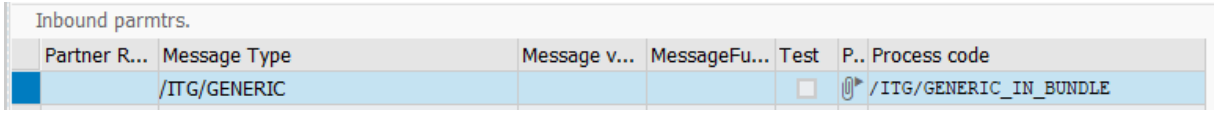
If the Data Replication for reference data objects shall be implemented using ALE the communication needs to be setup between the RDM system and the receiving systems.

3.9.2.1 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 4.1 Data Replication based on SAP ALE.
BD64 (example)



WE20 (example from receiving system)



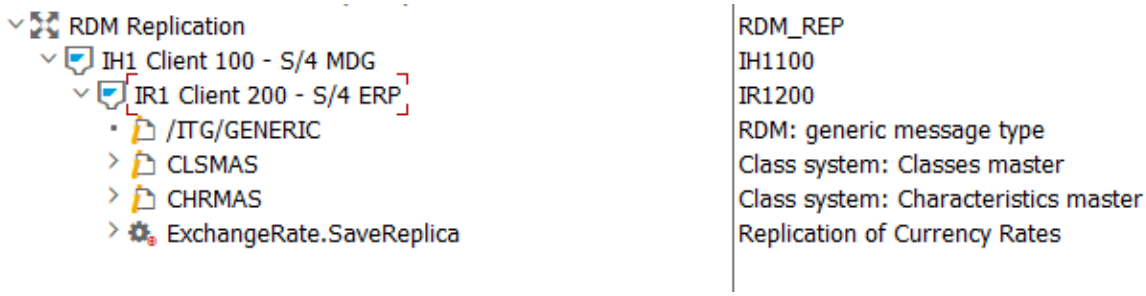
3.9.2.2 ALE Configuration for Object Specific Message Types

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:

Outbound

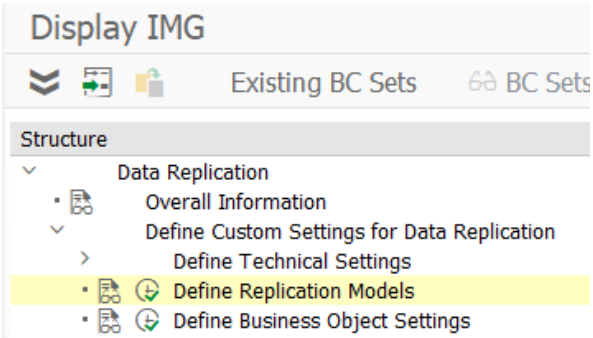
Partner R...	Message type	Message v...	Function	Test	Receiver ...	I...	Pa...	Basic type
	CHRMAS				A000000001		100	CHRMAS05
	CLSMAS				A000000001		100	CLSMAS04
	EXCHANGE_RATE				A000000001		100	EXCHANGE_RATE01

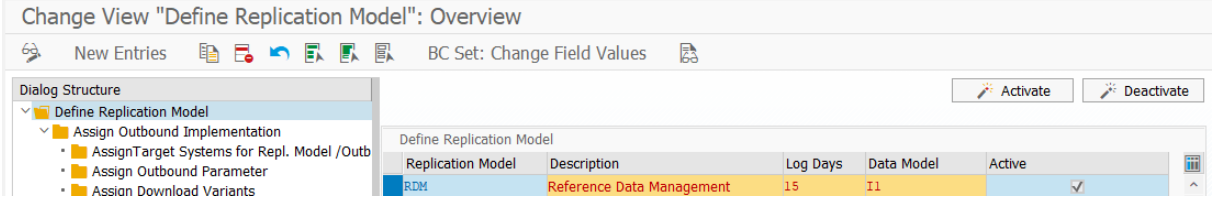
Transaction WE20 (Receiver), after generation of partner profiles:

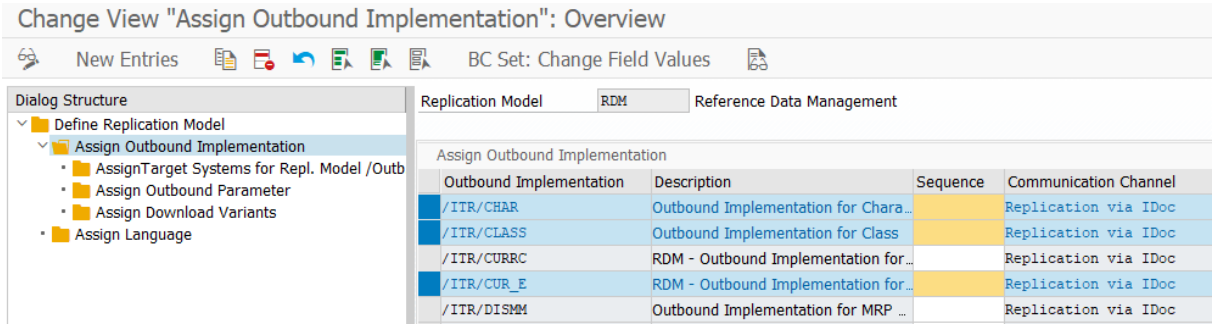
Inbound

Partner R...	Message Type	Message v...	Function	Test	P..	Process code
	CHRMAS					CHRM
	CLSMAS					CLSM
	EXCHANGE_RATE					BAPP

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







3.9.3 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.

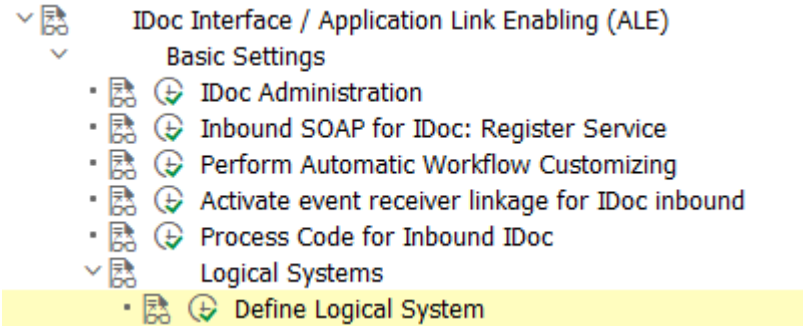
The web services offer two functionalities:

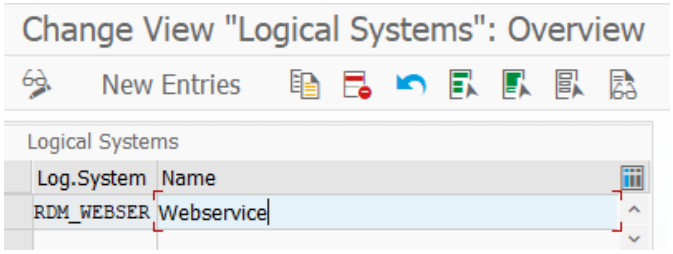
- Push (Send from RDM)
- Pull (Call from outside)

Examples:

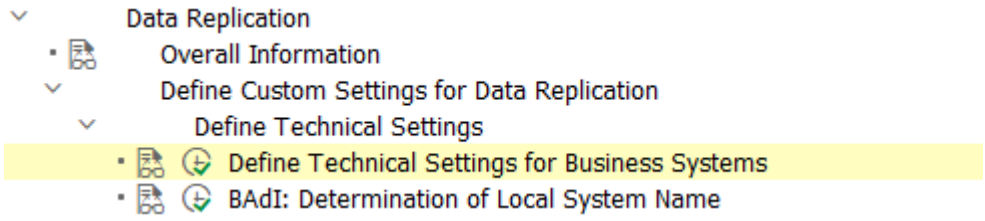
- Push (generic): /ITR/RDM_WS_SEND_OBJECT
- Pull (generic): /ITR/RDM_WS_GET_OBJECT
- Pull (object-specific): /ITR/RDM_WS_GET_PLANT

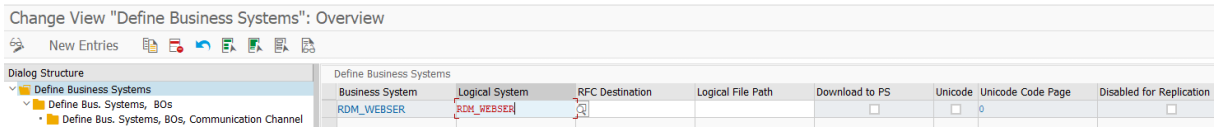
Transaction SALE: Define Logical System

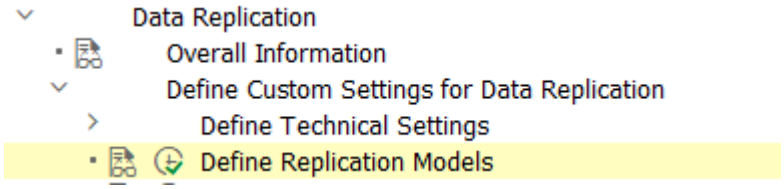


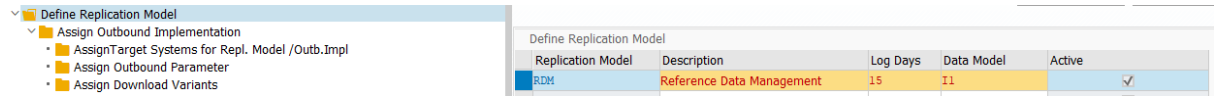


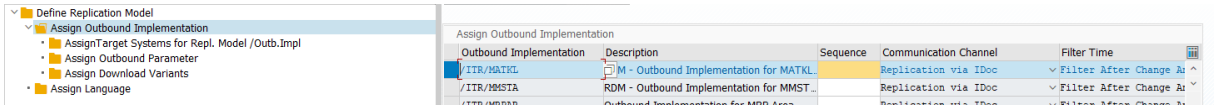
Transaction DRFIMG: Define Technical Settings and Replication Models

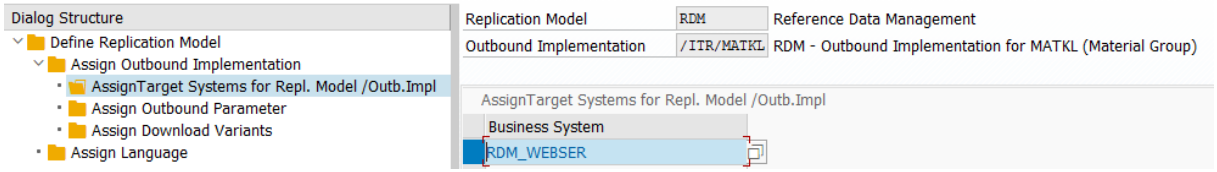


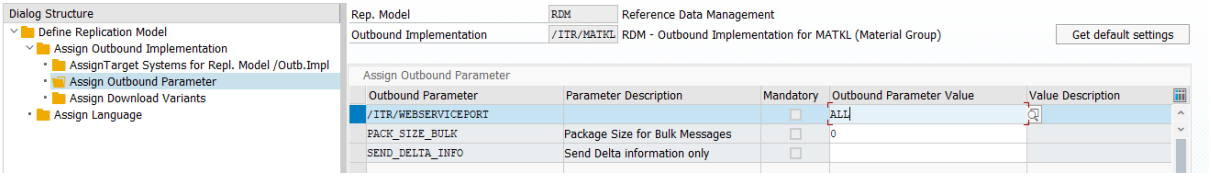






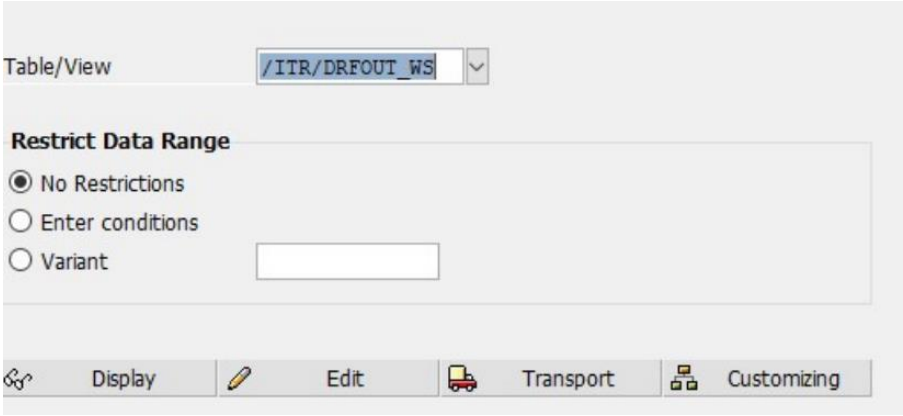


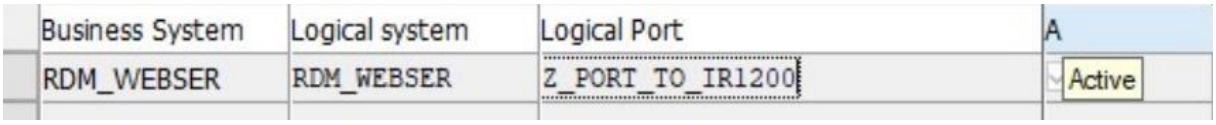




Using outbound parameter /ITR/WEBSERVICEPORT you can define which logical port will be used. If you define Value "ALL" all ports defined will be used.

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 4.2 Data Replication based on Webservices.

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

3.9.4 Key Mapping

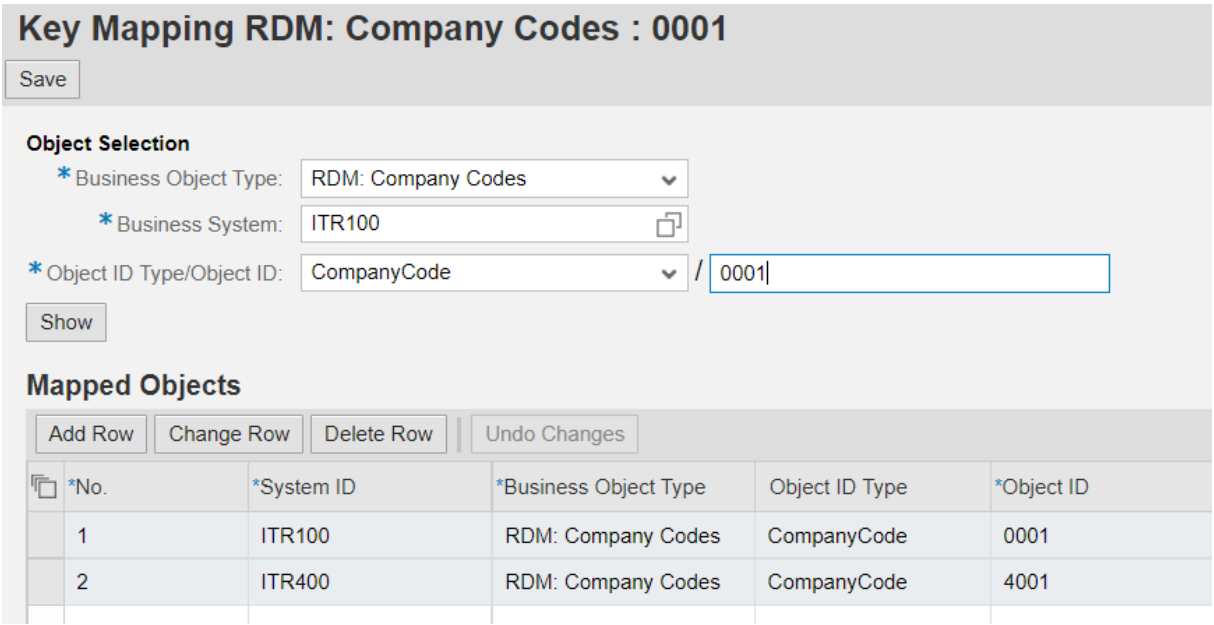
SAP MDG provides the following transactions to maintain and analyze Key Mapping:

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

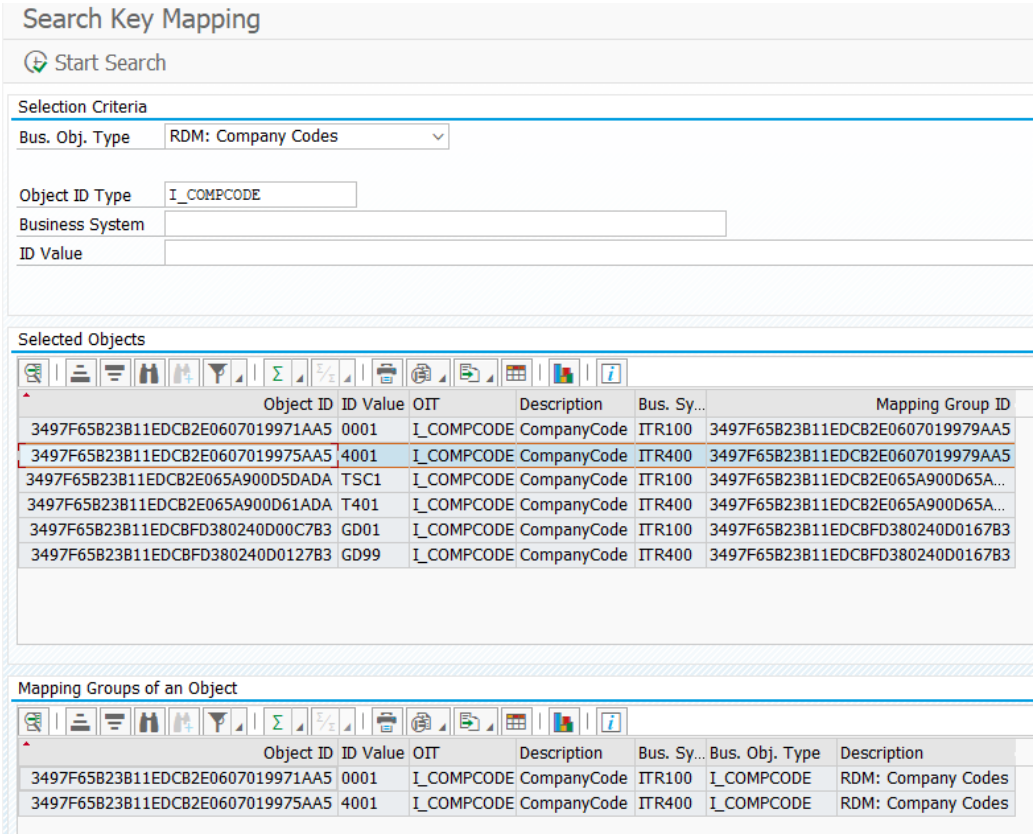
Use these transactions to maintain or search the Key Mapping for reference data object types by using:

- 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:



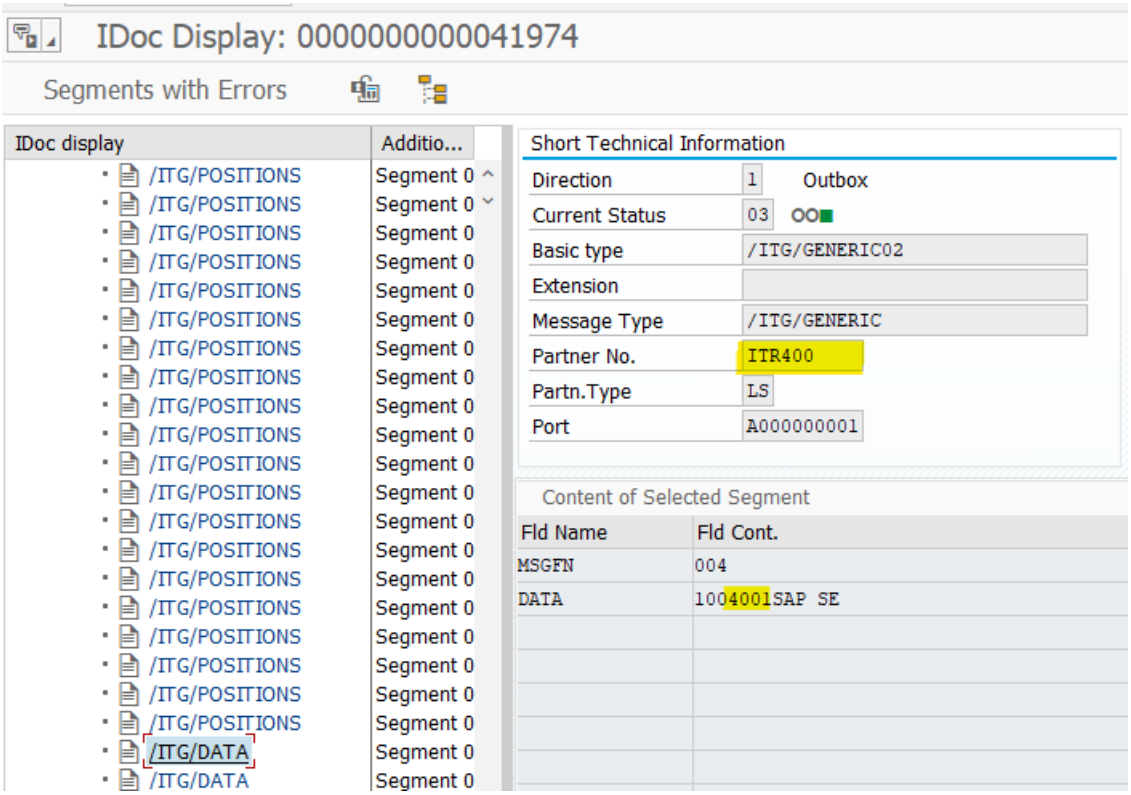
Example for Search Key Mapping:



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):



3.9.5 Value Mapping

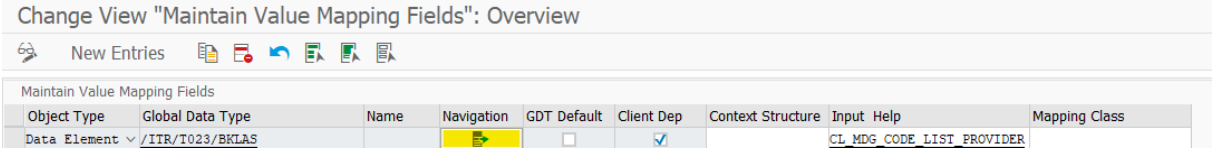
SAP provides the transaction "VMIMG - Value Mapping Customizing" to maintain Value Mappings. Use this transaction to maintain Value Mapping for reference data object types by using:

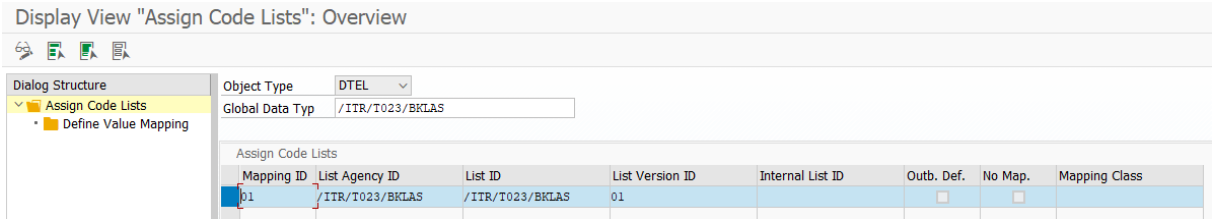
- Type: Data Element
- Global Data Type (GDT): "/ITR/<table>/<attribute>". Example: "/ITR/T023/BKLAS" (Valuation Class for Material Group).

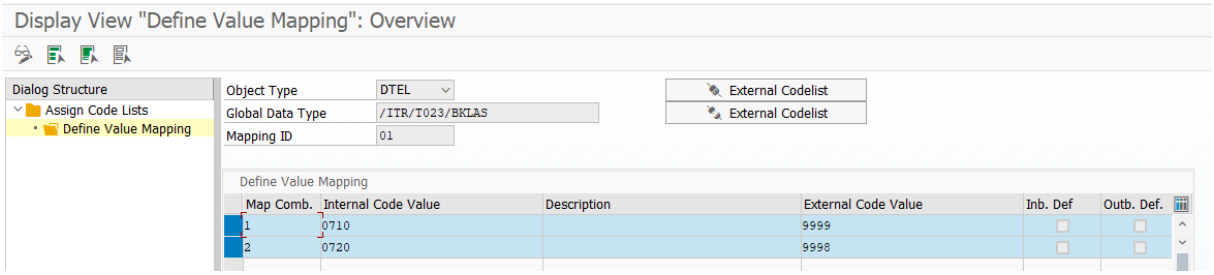
Examples for other attributes:

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

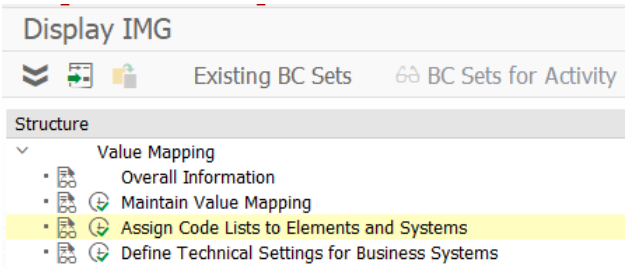
Maintained example:

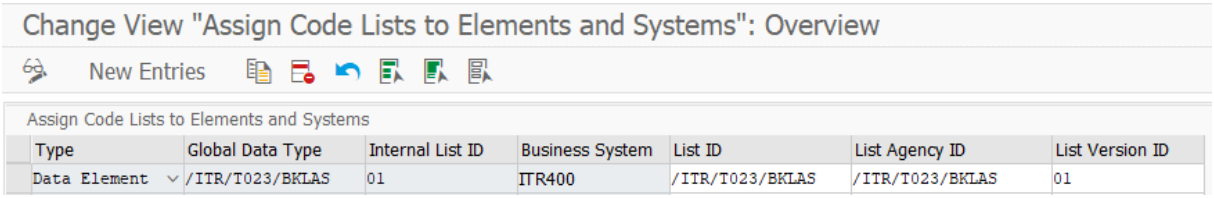






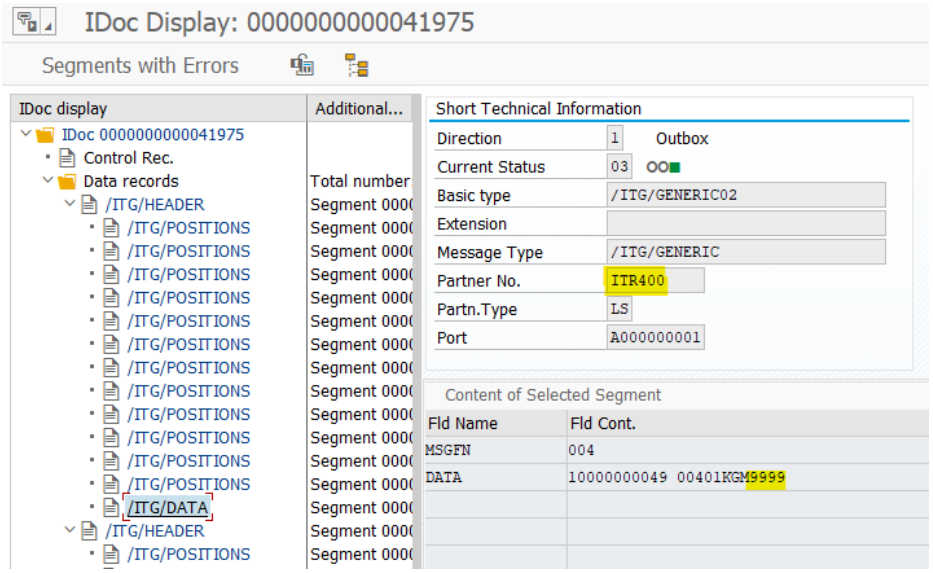
External codes (for system ITR400) for internal codes. Example: 9999 for 0710.



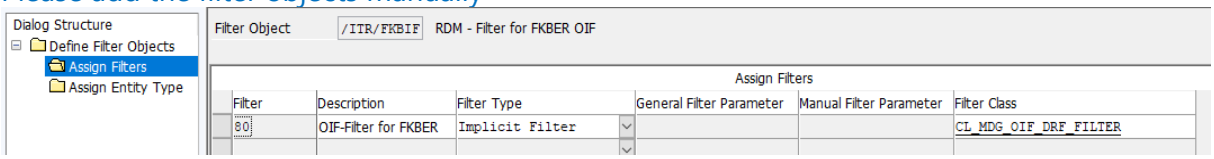


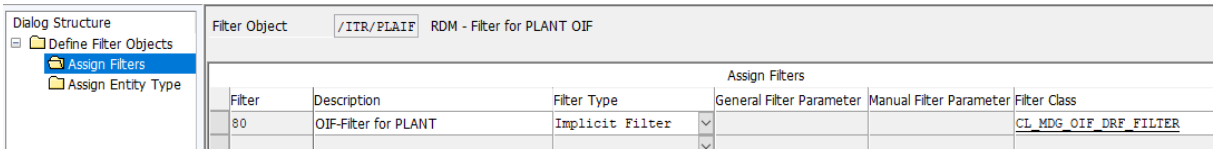
Code list are now in this example assigned for Business System ITR400. Make sure that this entry is added after you added the mapping in step “Maintain Value Mapping”. Otherwise the system will not allow you to add this entry here.

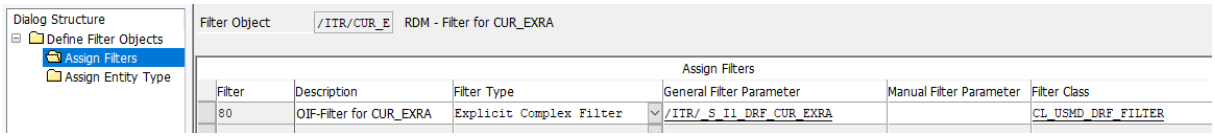
In the data replication the maintained values will be mapped based on the receiver system. Example, based on the maintained mappings above (transaction BD87):



Please add the filter objects manually







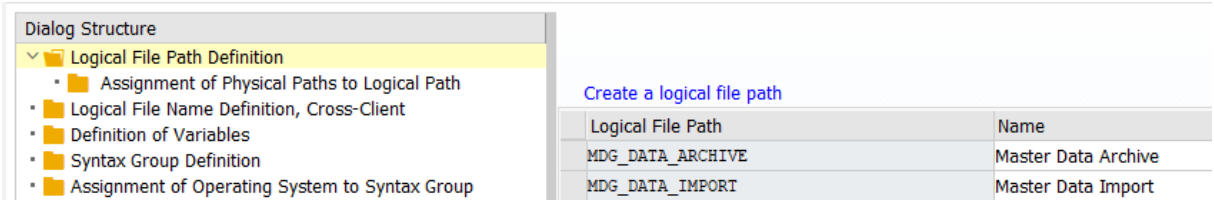
3.10 Set Up Data Transfer

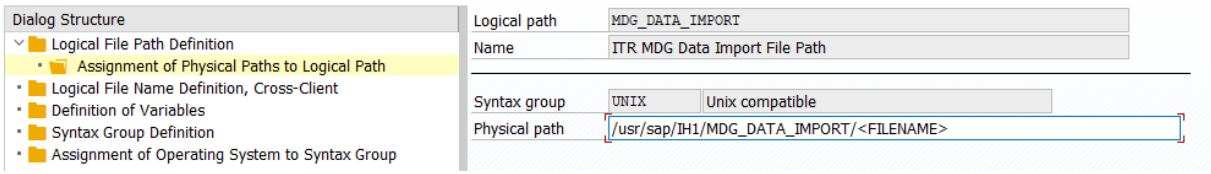
Data Transfer needs to be configured for initial load and consist of the configuration of MDMGX (on an SAP business system) and Data Import (on the MDG RDM system).

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.

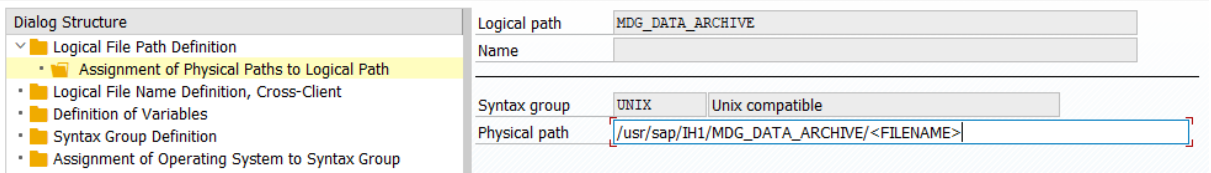
Please also check section 3.3 "Activate Business Configuration Sets" for the activation of the Data Transfer on the MDG system (BC Sets /ITR/MDG_RDM_DT_<n>). These BC Set deliver the necessary object type definitions. After this configure Data Transfer in transaction MDGIMG: Data Transfer -> Define File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

Transaction FILE:

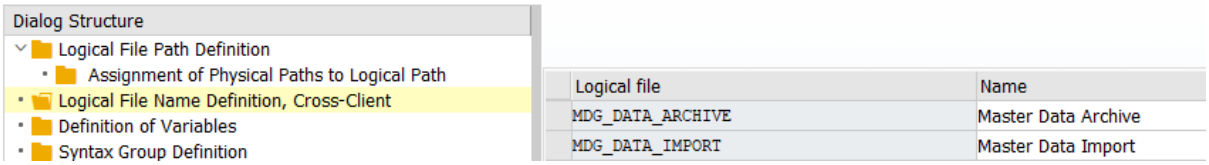


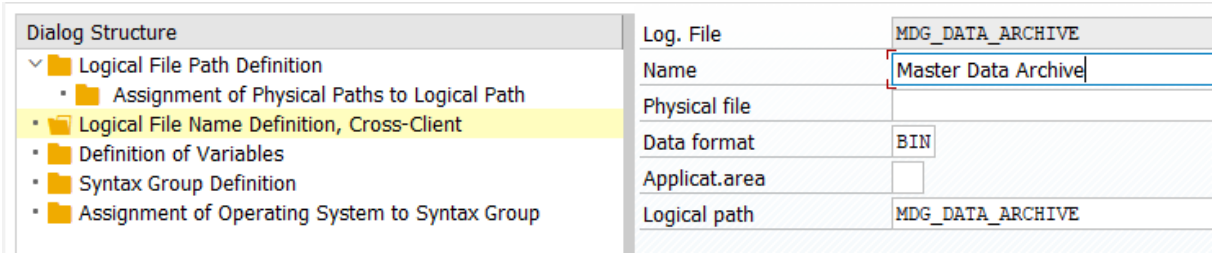


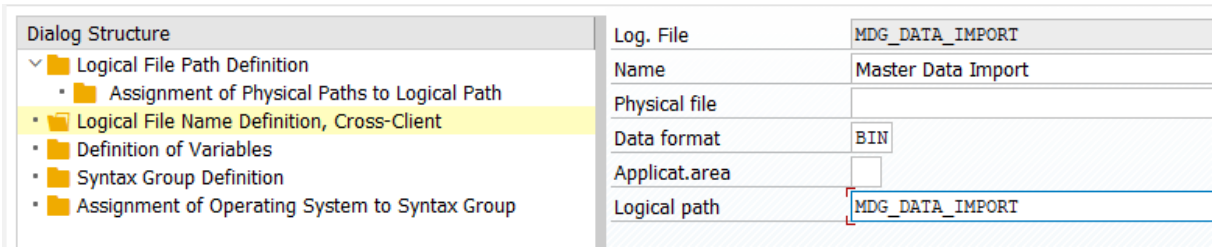
e.g.: /usr/sap/<systemID>/MDG_DATA_IMPORT/<FILENAME>



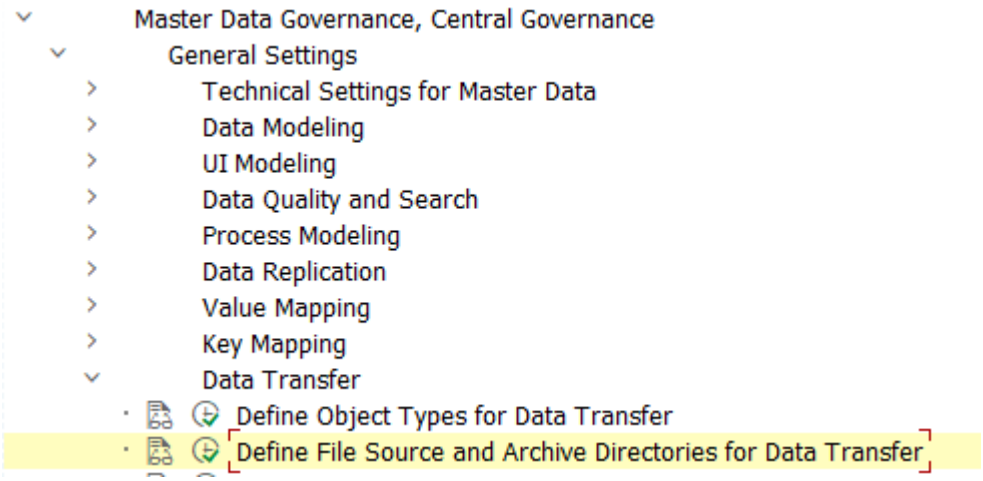
e.g.: /usr/sap/<systemID>/MDG_DATA_ARCHIVE/<FILENAME>

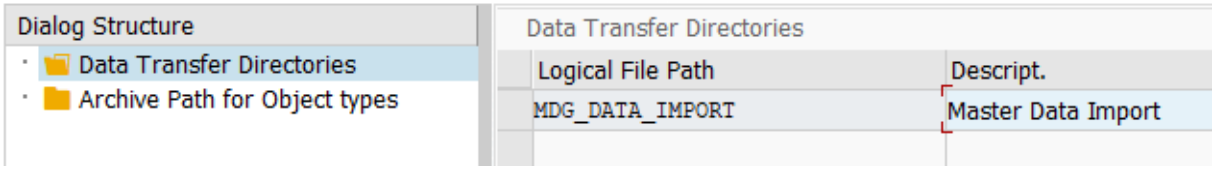


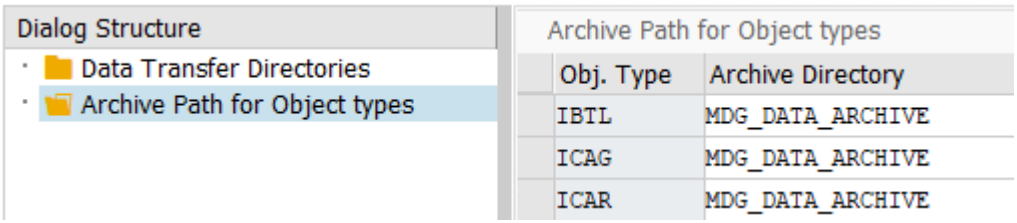




Using transaction MDGIMG configure the usage of these directories:







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.

3.11 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:

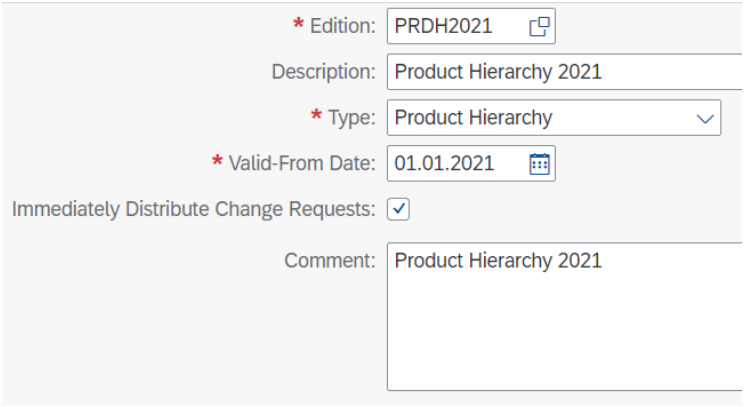
- Verify the product hierarchy usage in your business applications
- Check Interlocking
- Define Edition
- Creation of a Product Hierarchy Name
- Definition of Product Hierarchy Levels (e.g. "Branch" or "Division")
- Maintenance of number ranges

Verify the product hierarchy usage: 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. In order to obtain more information about the configuration of a Product Hierarchy contact: support@itego.de - Subject: "RDM Product Hierarchy".

Check Interlocking: Execute transaction MDGIMG: 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.

For the Edition definition start the NetWeaver Business Client and select Analysis of Editions -> Create. Use Edition Type "Product Hierarchy (I1_PRODH)" and define the Data Replication Timing "On Final Approval of Change Request" (Immediately Distribute Change Requests).

Example:



Note: parallel editions are currently not supported

The creation of the Product Hierarchy Name is done through a change request process. Start the NetWeaver Business Client 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.

Define Levels for the Product Hierarchy using transaction SM30: Select /ITR/PRODH_LVL and define the levels according to your usage scenario. E.g.:

- 1 Branch
- 2 Division

In the standard delivery the first two levels are represented by Product Hierarchy Nodes and supplemented by Level 3 represented by Product Hierarchy Sub Nodes which do not have to be configured in customizing view /ITR/PRODH_LVL.

The maintenance of number ranges starts with definition of the "From No." and the "To Number" for each level of the Product Hierarchy Nodes and the 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 (internal numbering):

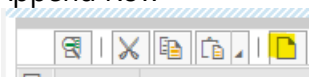
- No: 01; From No: 0000000000000001; To No: 0000000000049999
- No: 02; From No: 0000000000050000; To No: 0000000000099999
- No: 03; From No: 0000000000100000; To No: 0000000000999999

The number ranges have to be maintained on the MDG and on the SAP business application system.

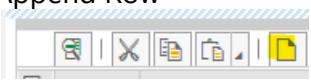
On the SAP business application system execute transaction SA38 and choose report /ITR/MDG_PRODH_EXPORT to extract the product hierarchy using the defined number ranges which have been configured for MDG RDM. Choose the following parameters:

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

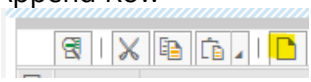
- Append Row



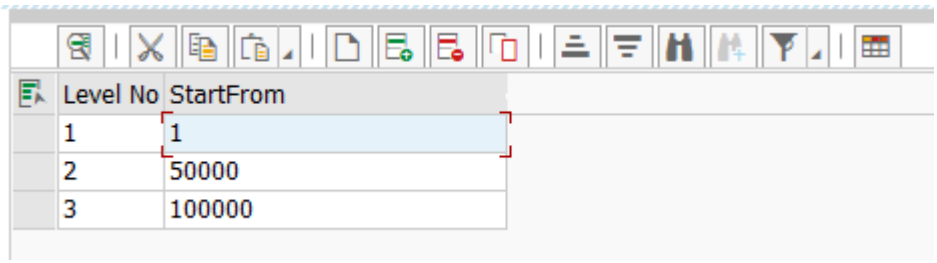
  - Level: 1; <StartFrom> (any number which fits to the number range defined above for range number 01; e.g. 00001 for the initial load)
- Append Row



  - Level: 2; <StartFrom> (any number which fits to the number range defined above for range number 02; e.g. 50000 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. 00100000 for the initial load)



Level No	StartFrom
1	1
2	50000
3	100000

For a standard three level hierarchy e.g. also the following entries might be a good example:

- 1: 000000000000000001
- 2: 000000000000100000
- 3: 000000010000000000

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

- 1: 000000000000000001
- 2: 000000000000000100
- 3: 000000000000100000
- 4: 000000001000000000
- 5: 000010000000000000

Note: The number of characters which can be used for <StartFrom> for Level 1-n is defined by structure PRODHS in your local SAP business system. In the standard delivery this is defined as a character field with length 18, divided into 5, 5 and 8 characters for level 1-3. This is why in the example above the levels to be configured are defined by 5, 5 and 8 characters (unlike the number ranges in the MDG RDM system, which represent the same numbers but do have 18 characters for each level.

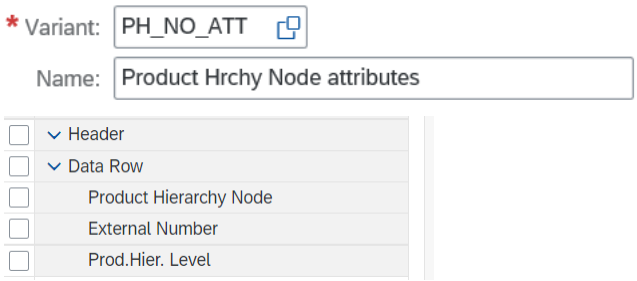
After the extraction the files have to be loaded to the MDG RDM system using File Upload. This should be done in the following sequence for a standard product hierarchy:

- Node Level 1 attributes and texts
- Node Level 2 attributes and texts
- Sub Node 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 sub nodes to level 2 nodes

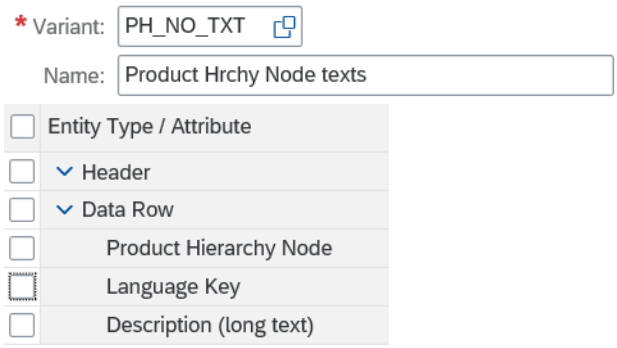
Please verify that all numbers have been generated according to the number ranges defined.

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

- Node 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



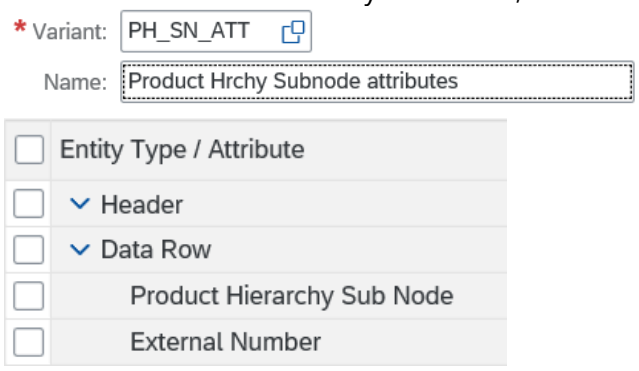
  - Texts Data Row: Product Hierarchy Node, Language Key, Description (long text)



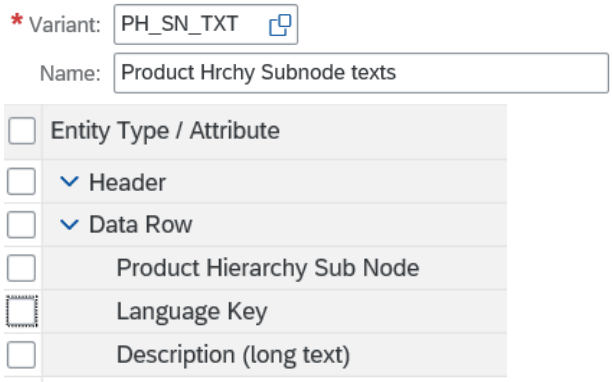
- Node Level 2 attributes and texts
  - Same Entity Type and variants as for Level 1 Nodes

- Sub Node attributes and texts (PH_SN_ATT and PH_SN_TXT)
  - Entity Type: Product Hierarchy Sub Node

- Attributes: Product Hierarchy Sub Node, External Number

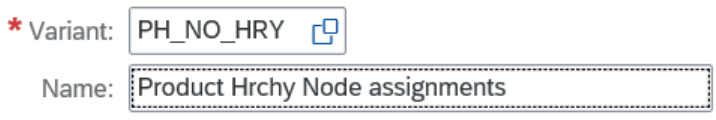


- Texts: Product Hierarchy Sub Node, Language Key, Description (long text)

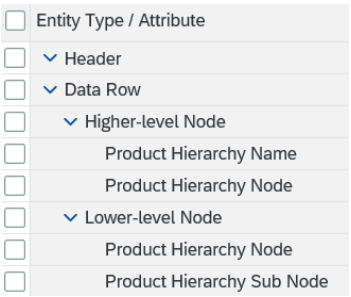


- Hierarchy assignments
  - Entity Type: Product Hierarchy Node
  - Higher-level Node: Product Hierarchy Name, Product Hierarchy Node
  - Lower-level Node: Product Hierarchy Node, Product Hierarchy Sub Node

- For an "all-in-one upload" you can use this variant definition:



-



- 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 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

- Level 3 hierarchy assignments of sub nodes to level 2 nodes
  - Entity Type: Product Hierarchy Node
  - Higher-level Node: Product Hierarchy Name, Product Hierarchy Node
  - Lower-level Node: Product Hierarchy Node, Product Hierarchy Sub 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>

Load the following files:
- Node Level 1 attributes and texts
  - Attributes: PRODH_EXTRACT_1_*
  - Texts: PRODH_EXTRACT_1_T*

- Node Level 2 attributes and texts
  - Attributes: PRODH_EXTRACT_2_*
  - Texts: PRODH_EXTRACT_2_T*

- Sub Node attributes and texts
  - Attributes: PRODH_EXTRACT_3_*
  - Texts: PRODH_EXTRACT_3_T*

- Level 1 hierarchy assignments of level 1 nodes to the product hierarchy name)
  - Assignments: PRODH_EXTRACT_1_H*

- Level 2 hierarchy assignments of level 2 nodes to level 1 nodes
  - Assignments: PRODH_EXTRACT_2_H*

- Level 3 hierarchy assignments of sub nodes to level 2 nodes
  - Assignments: PRODH_EXTRACT_3_H*

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

In a last step the number range status for each number range has to be defined. Use transaction SNRO (Object: /ITR/PRODH) and choose "Interval Editing" for the maintenance process:

Example:

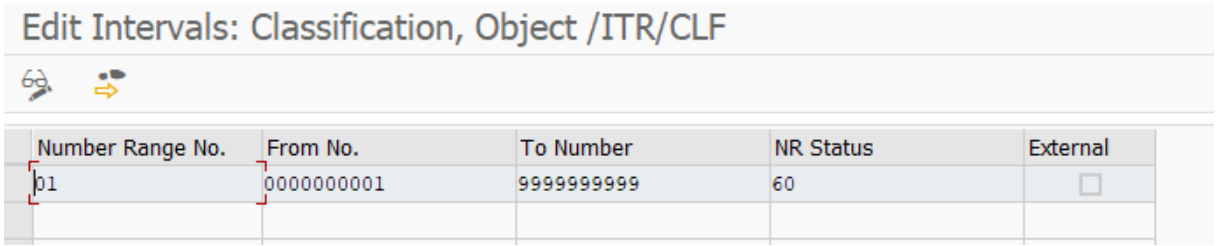
- No: 01; NR Status: 2; assuming 2 loaded level 1 nodes (1-2)
- No: 02; NR Status: 50004; assuming 5 loaded level 2 nodes (50000 – 50004)
- No: 03; NR Status: 100010; assuming 11 loaded sub nodes (100000 – 100010)

After this last step the maintenance of the product hierarchy can start and the next node and sub nodes create will get the next defined MDG number from the number range and the next external number defined by the external numbers based on the external numbers loaded from the SAP business system and the hierarchy assignments selected.

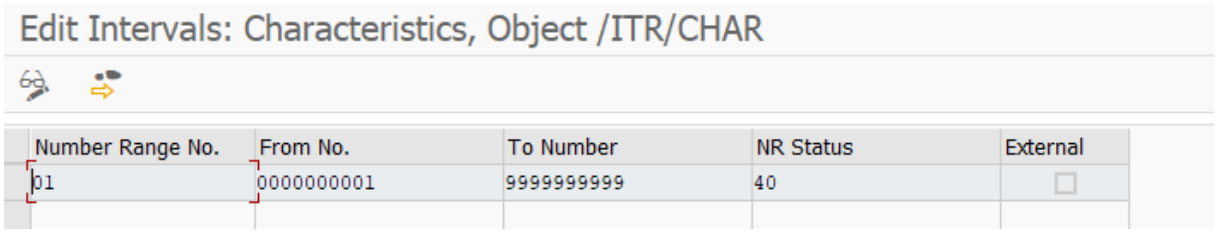
3.12 Configure Classification

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:

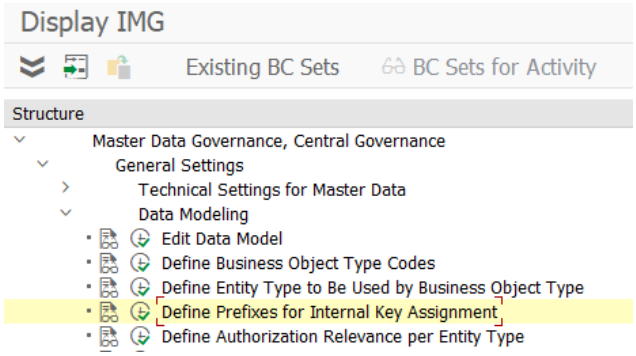
Classes:

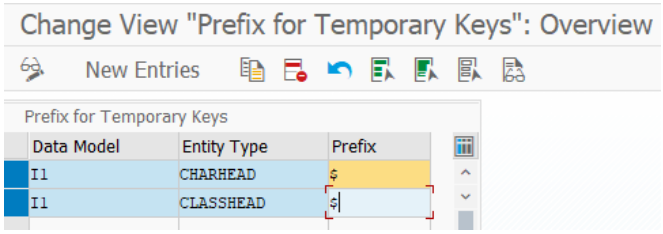


Characteristics:



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





3.13 Set Up Local Staging Areas

The configuration of the Local Staging Areas 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.

3.14 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".

3.15 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 Examples

4.1 Data Replication based on SAP ALE

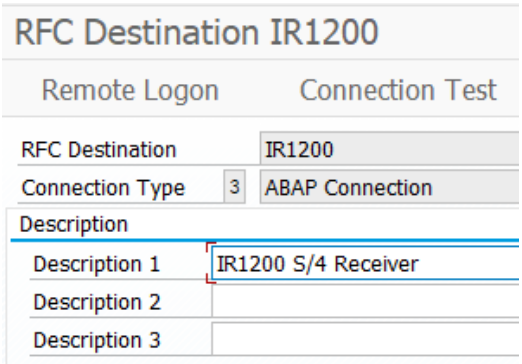
Most RDM objects will be replicated using a generic message type. This section shows how to set this up based on two systems:

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

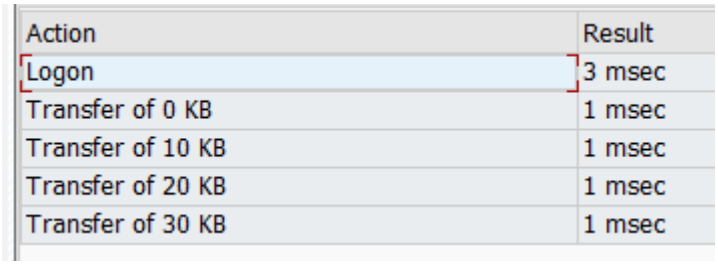
Please adjust the examples below to your own system landscape.

4.1.1 Prerequisite: RFC Destination

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



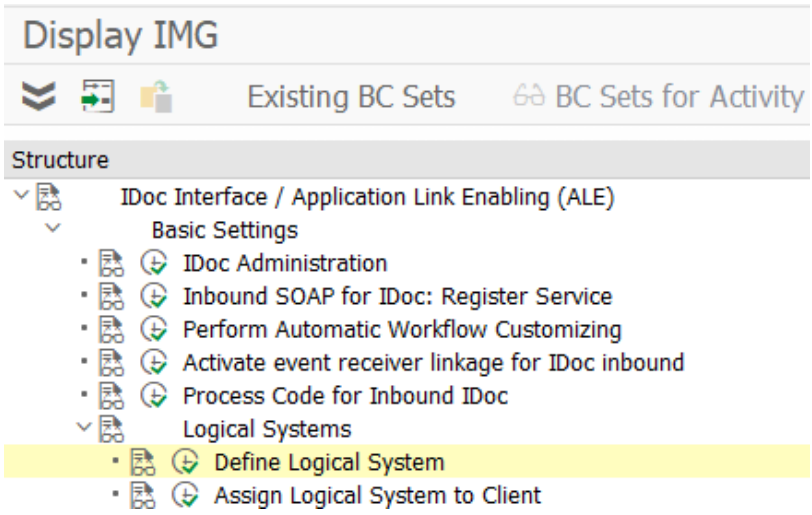
Connection Test should be successful

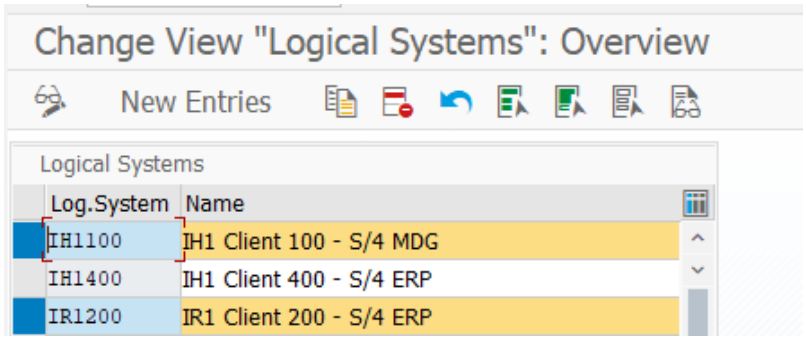


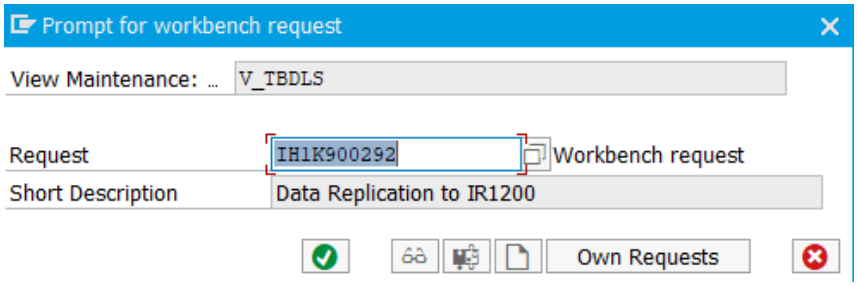
Action	Result
Logon	3 msec
Transfer of 0 KB	1 msec
Transfer of 10 KB	1 msec
Transfer of 20 KB	1 msec
Transfer of 30 KB	1 msec

4.1.2 Define Logical System

System IH1 100, transaction SALE:

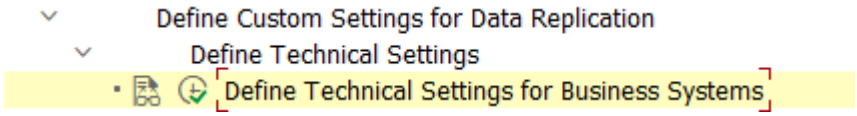


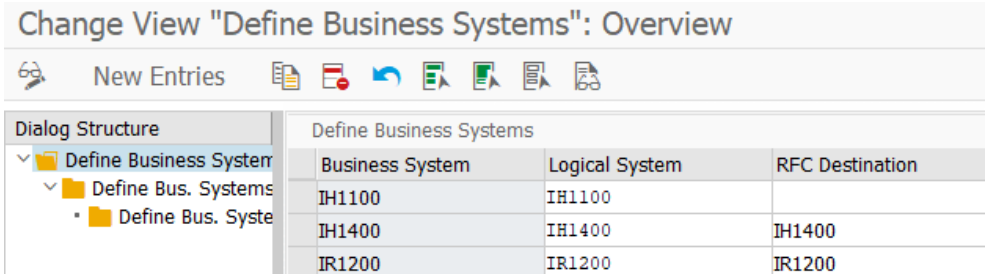




4.1.2 Check or Create Business System

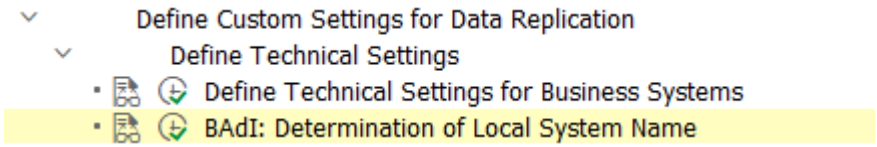
System IH1 100, transaction DRFIMG:



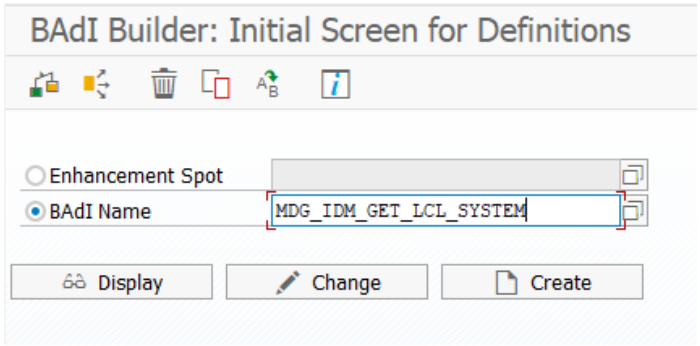


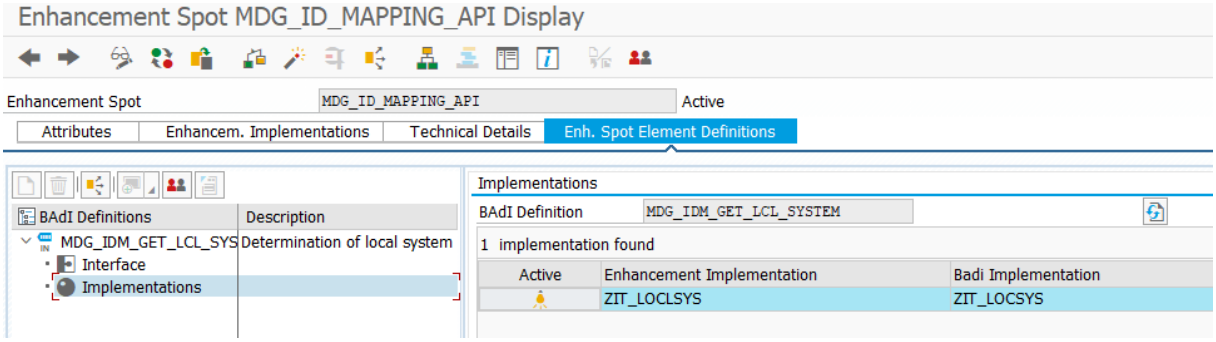
4.1.3 Check or Create BAdI for determination of local system name

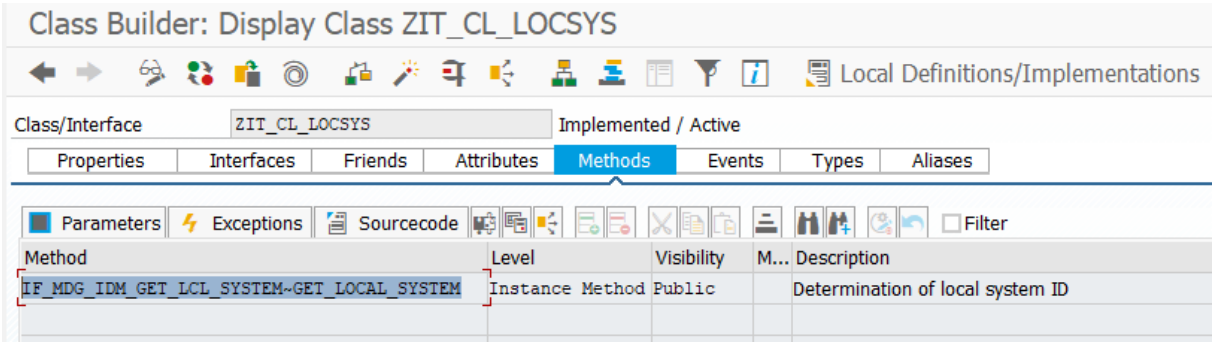
System IH1 100, transaction DRFIMG:



Or use transaction SE18, BAdI Name: MDG_IDM_GET_LCL_SYSTEM



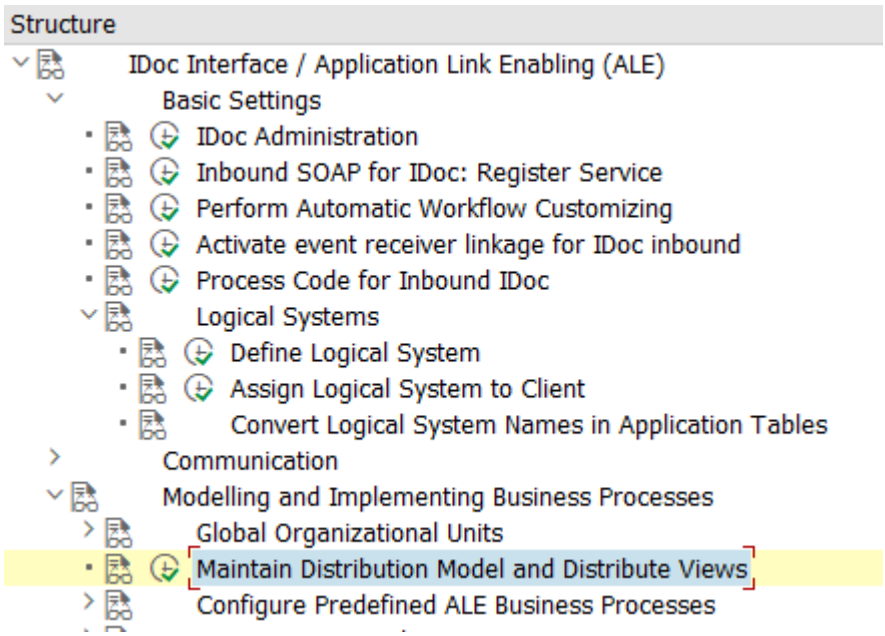




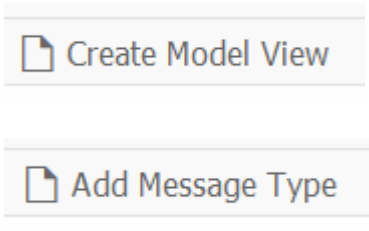

```
method IF_MDG_IDM_GET_LCL_SYSTEM~GET_LOCAL_SYSTEM.
*! This method determines the local business system via the ALE logical sys
tem assigned to it.
* If no business system is maintained, it returns an empty value.

  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.
```

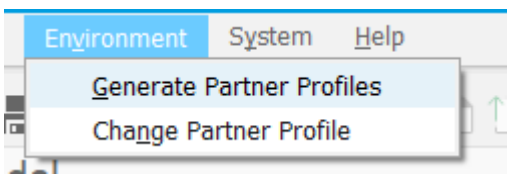
4.1.4 Define Distribution Model

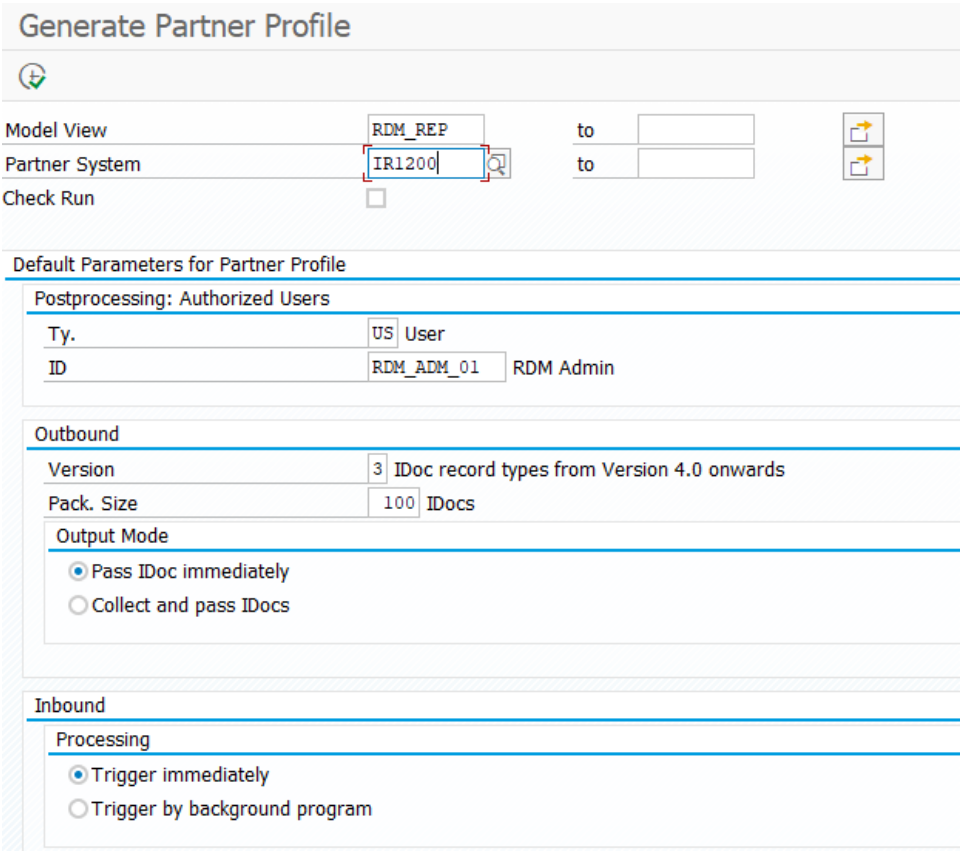
Use transaction SALE (or BD64):

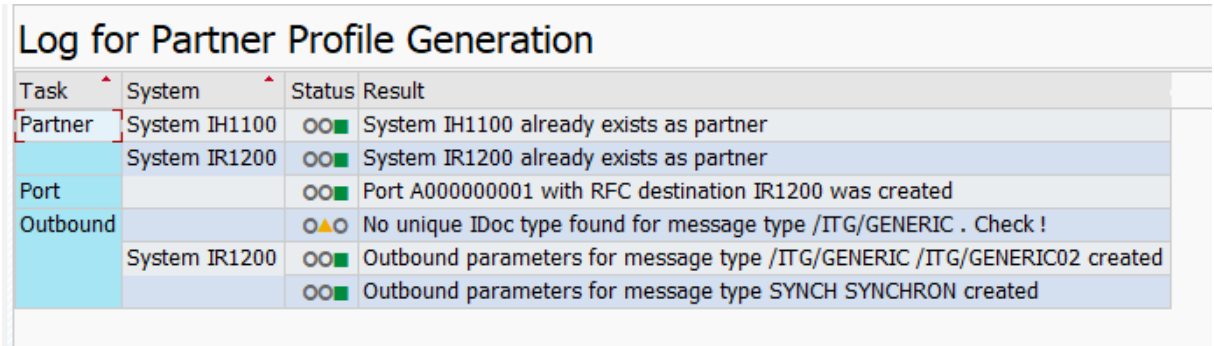


Create Model View and Add Message Type

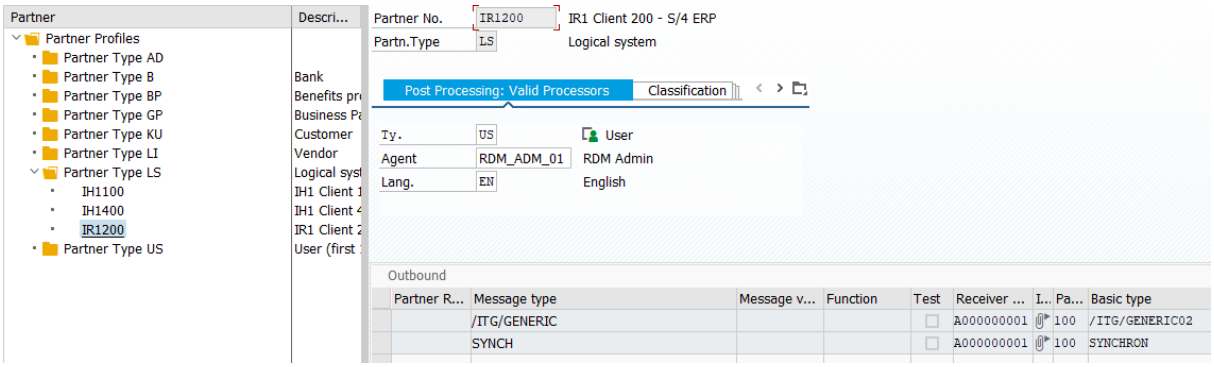




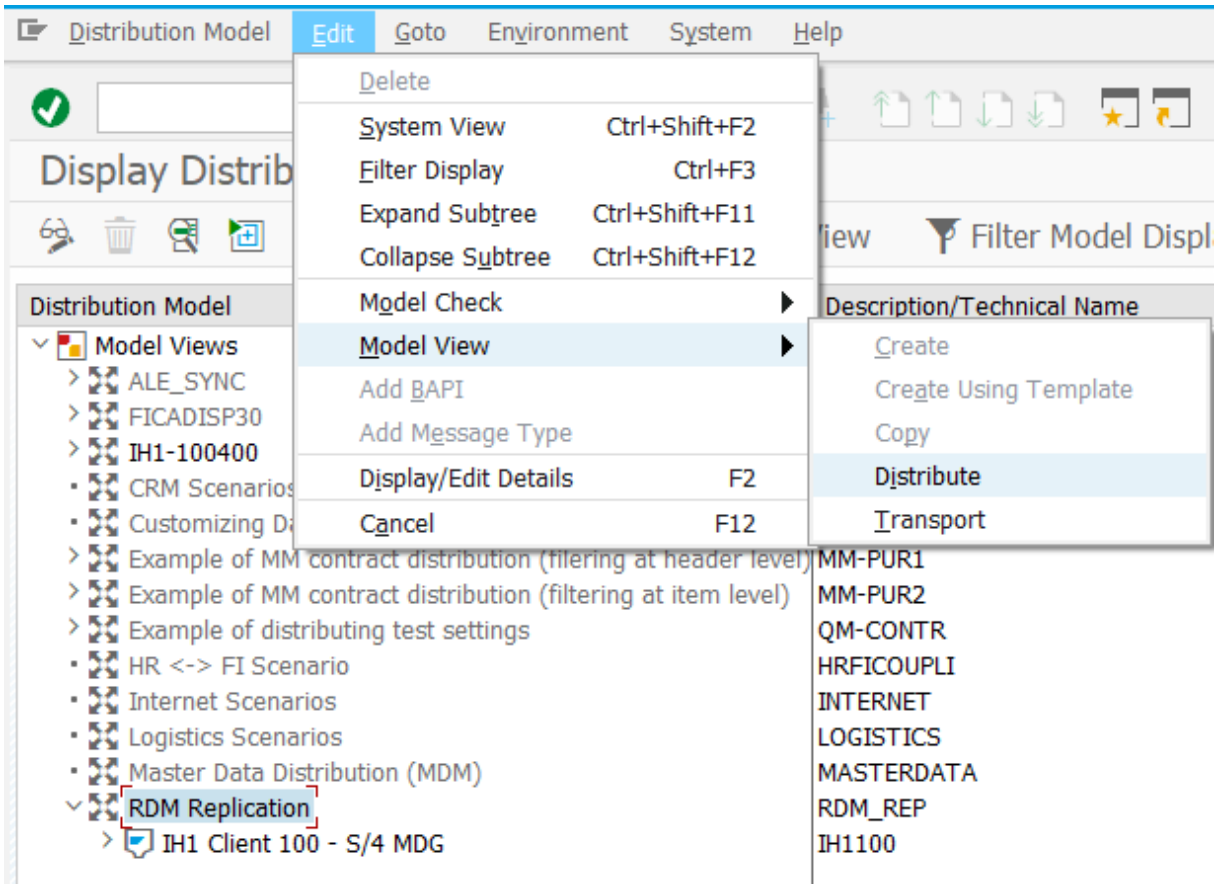


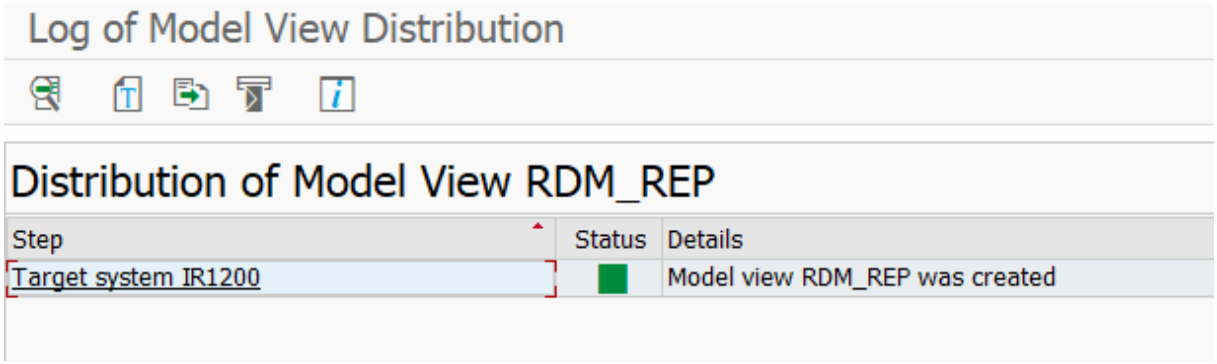


Check in transaction WE20:



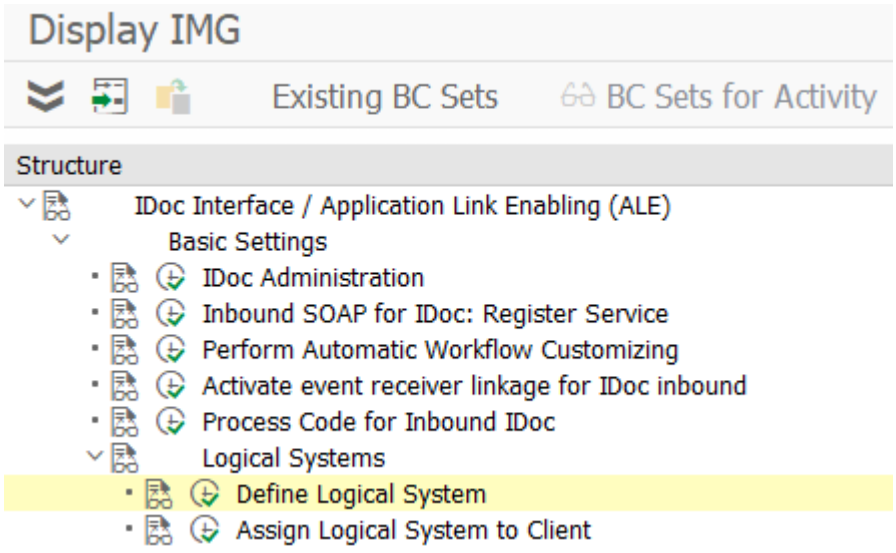
BD64: Distribute Model View to receiver (IR1200)

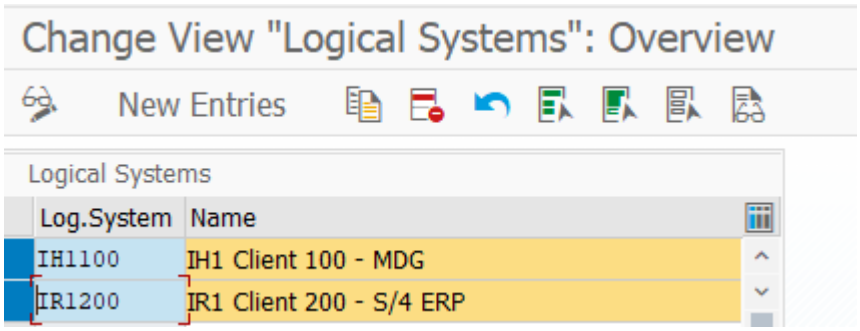


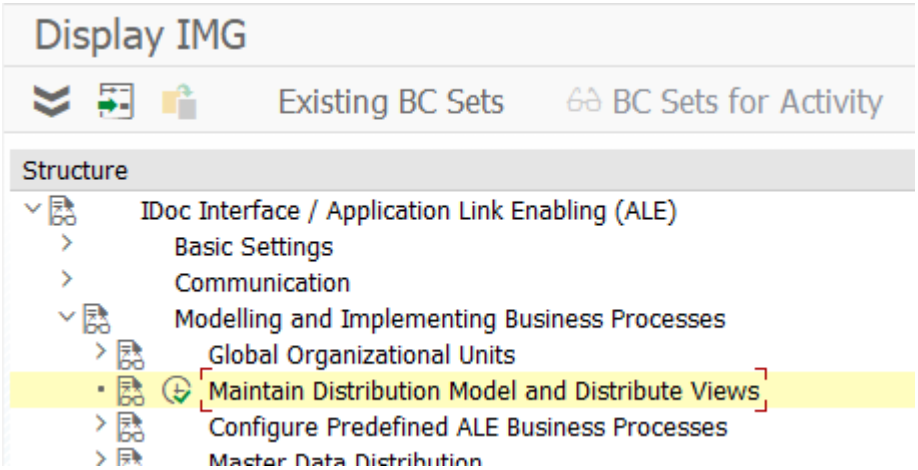


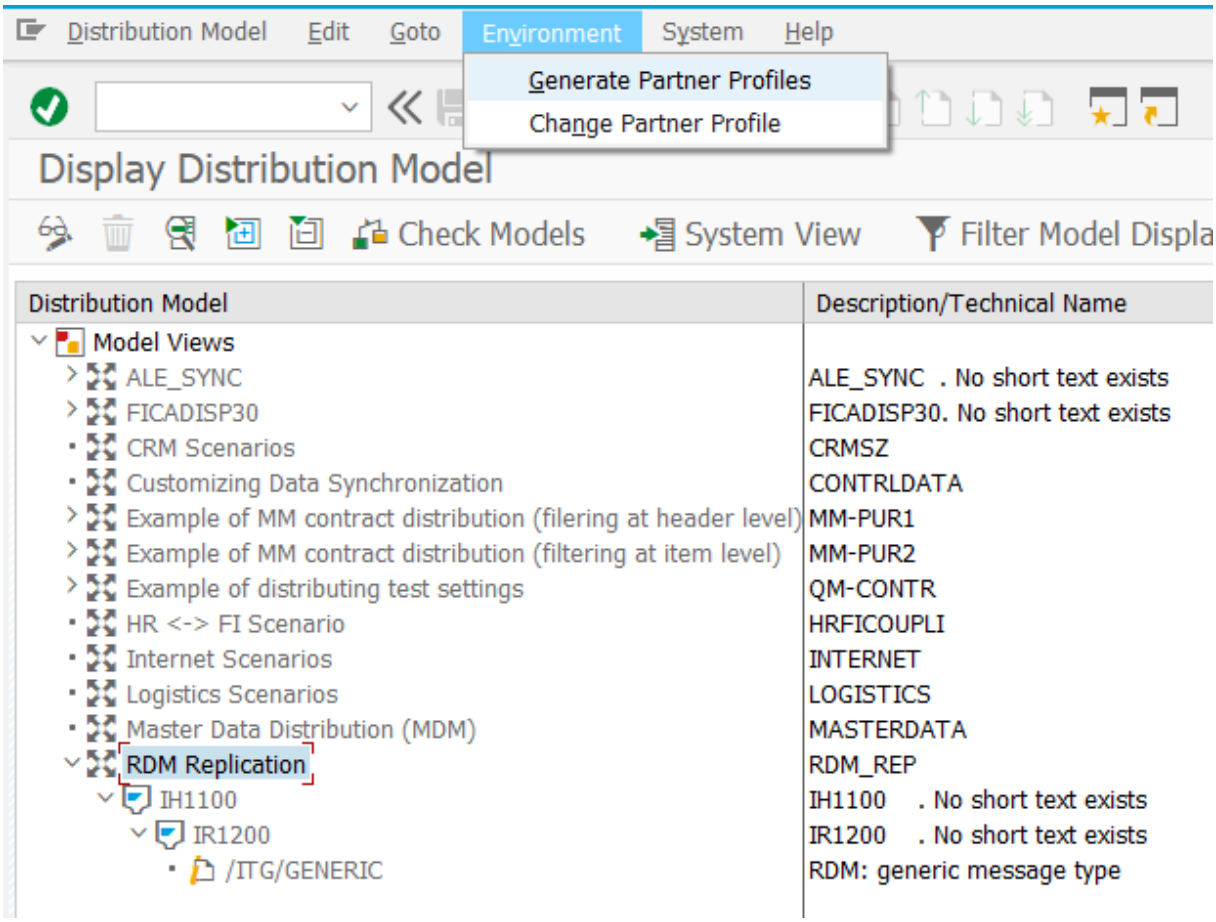
Log On to system IR1200 and execute transaction SALE:

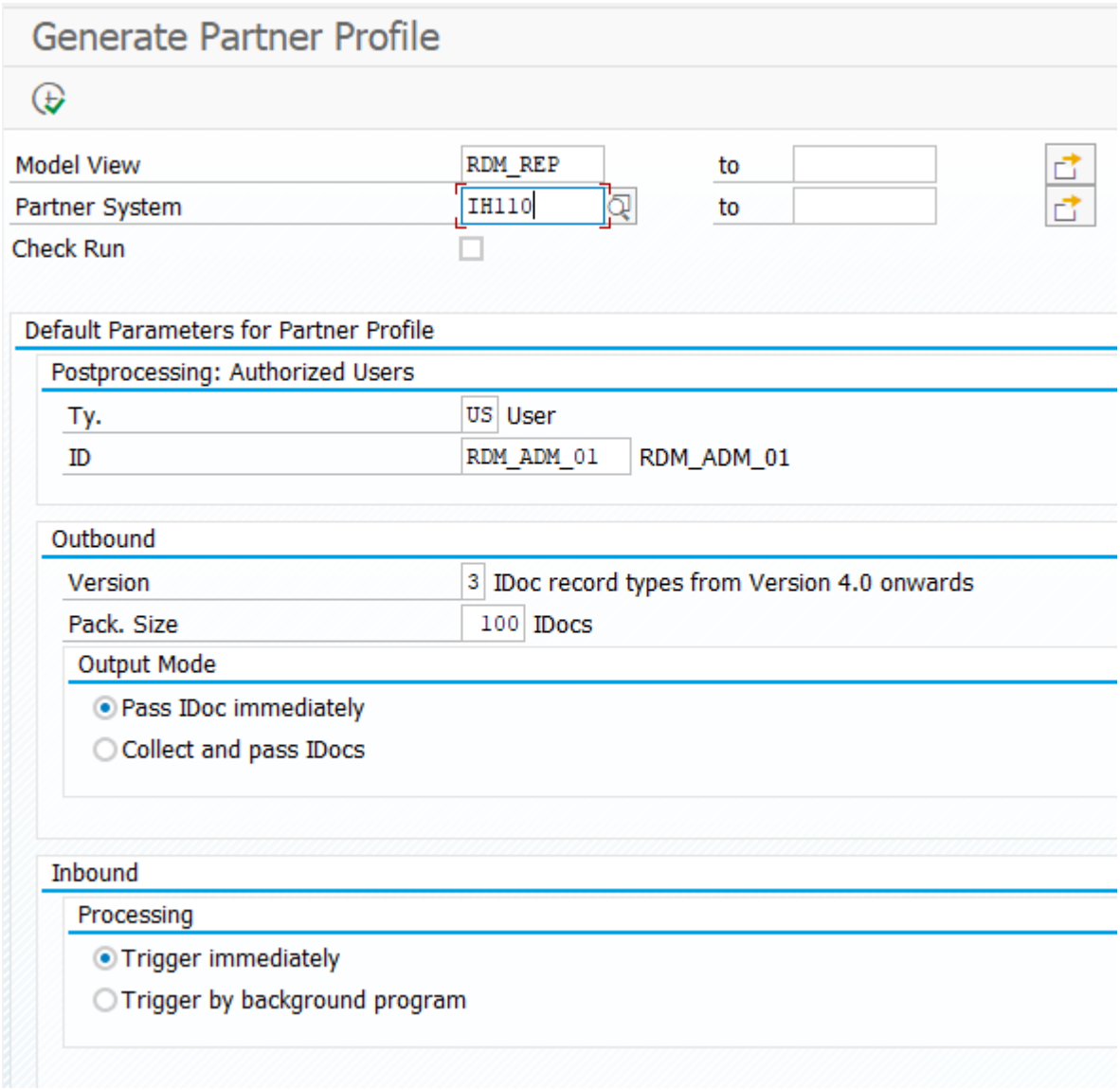
Check Logical Systems



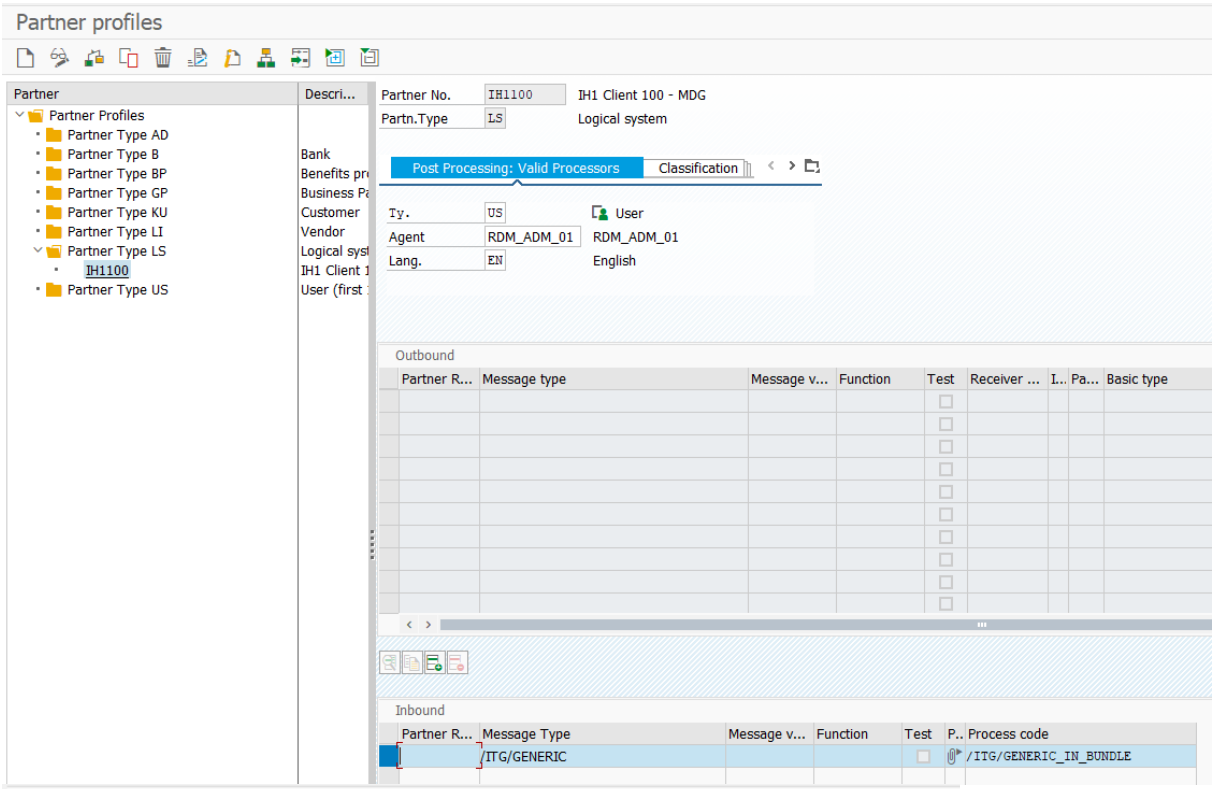


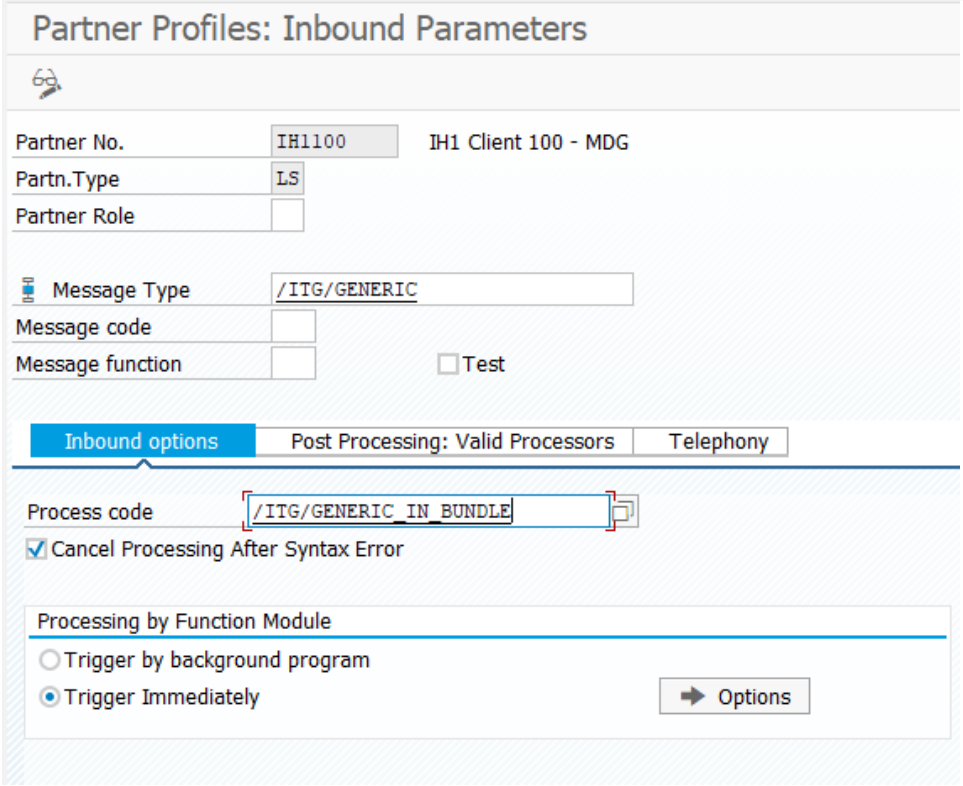

Display IMG
Existing BC Sets
BC Sets for Activity
Structure
IDoc Interface / Application Link Enabling (ALE)
Basic Settings
Communication
Modelling and Implementing Business Processes
Global Organizational Units
Maintain Distribution Model and Distribute Views
Configure Predefined ALE Business Processes


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

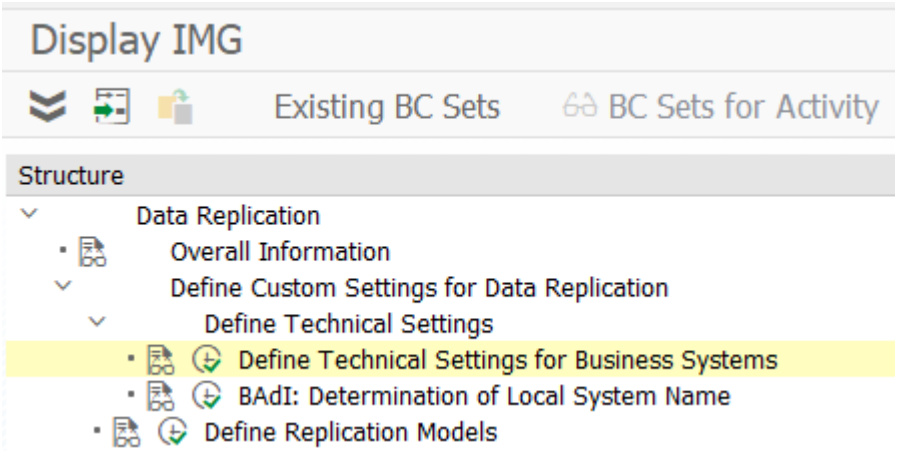


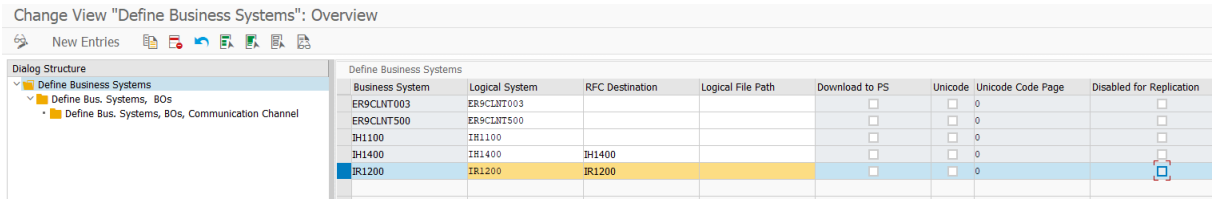
Check in WE20



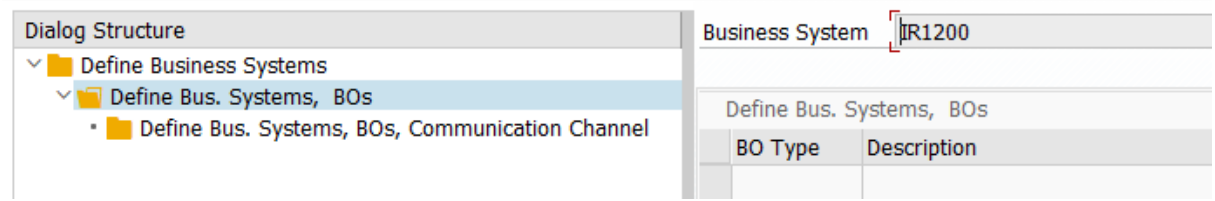


System IH1 100: Check DRFIMG:

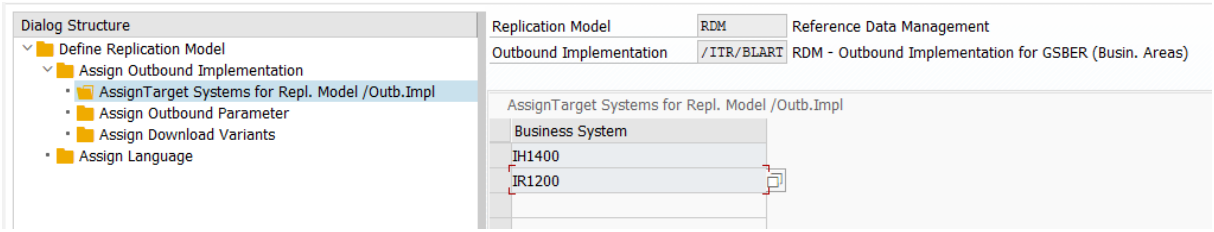


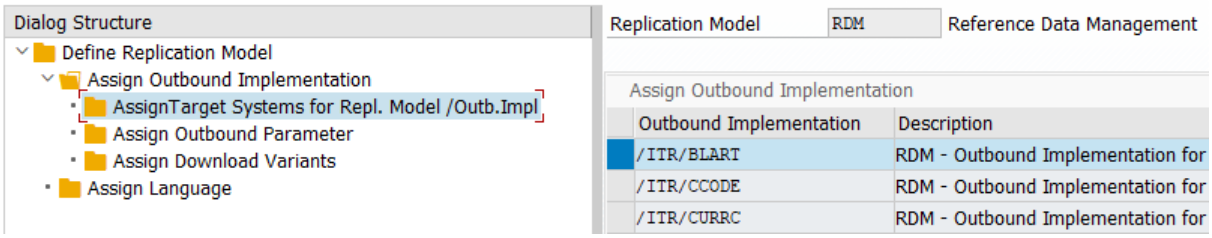


Nothing else to be configured:

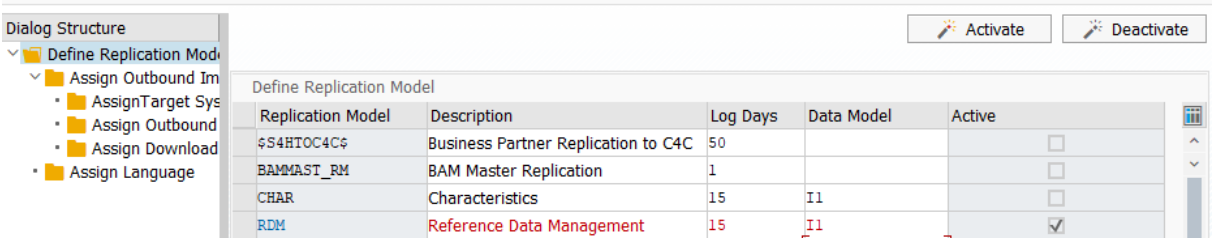


For every object in scope for this receiver system define target system and outbound implementation:





Data Replication Model needs to be active:



4.2 Data Replication based on Webservices

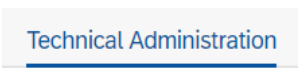
This section shows how to set up a Webservice based communication using the following systems:

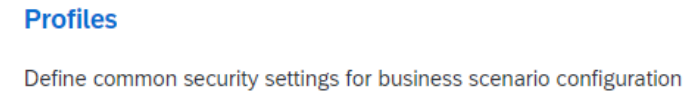
- RDM for MDG (sender): System IH1, client 100
- SAP Business System (receiver): System IR1, client 200

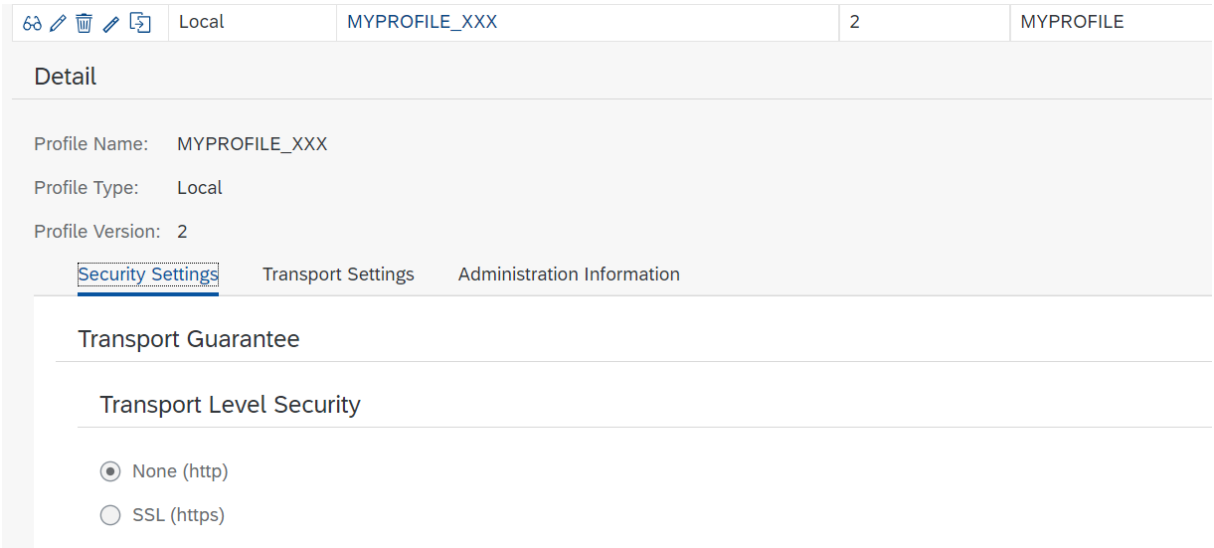
Please adjust the examples below to your own system landscape.

4.2.1 Receiver: Technical Administration: Profiles and Provider Systems

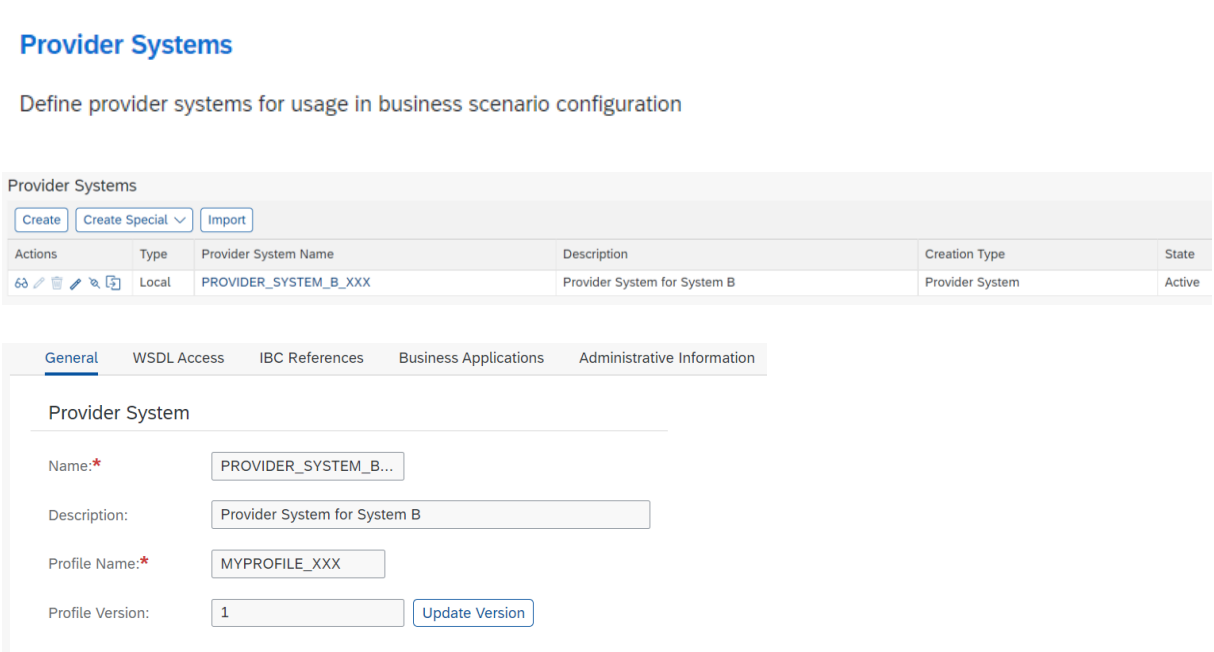
System IR1 200: transaction SOAMANAGER:

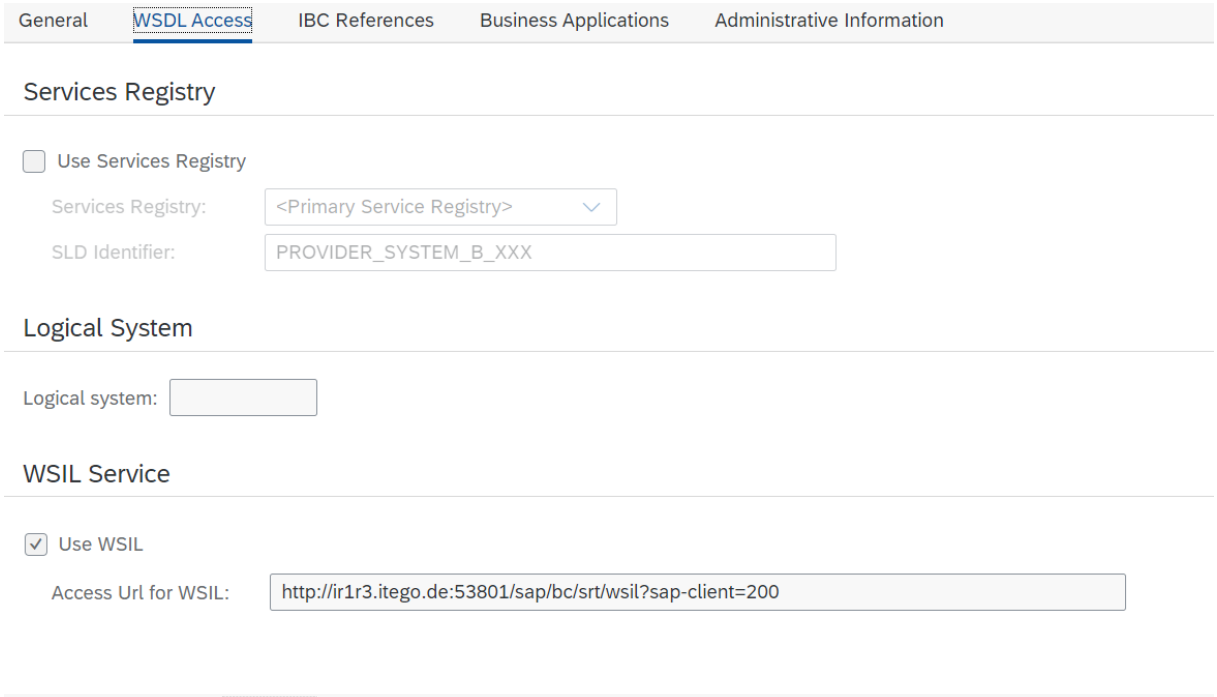


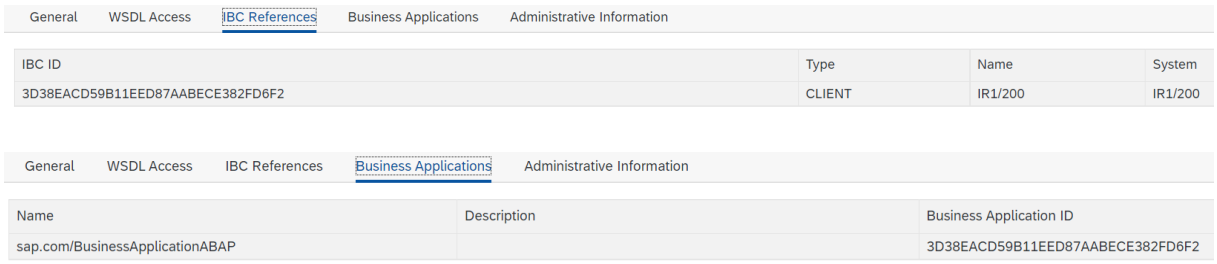




Or https based on your security requirements.

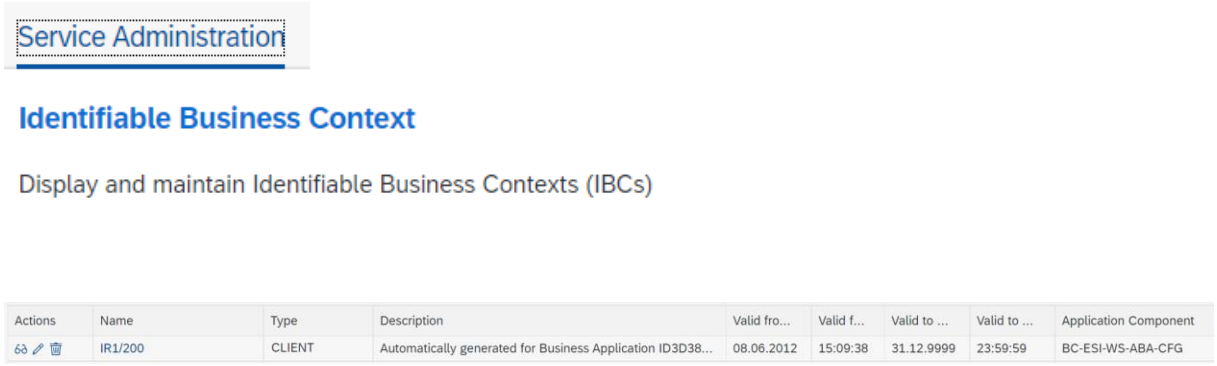


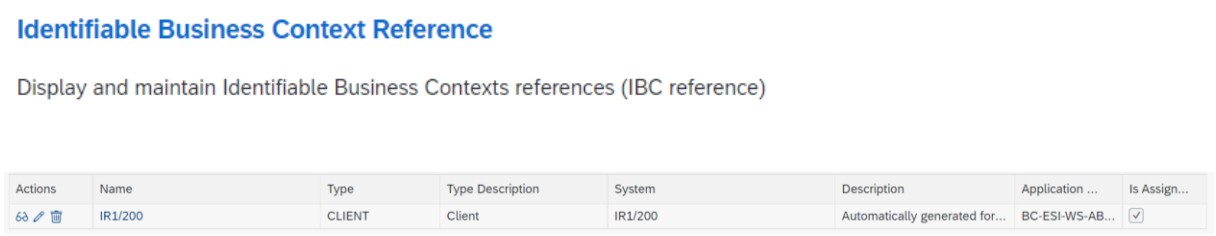




4.2.2 Receiver: Service Administration: Business Context

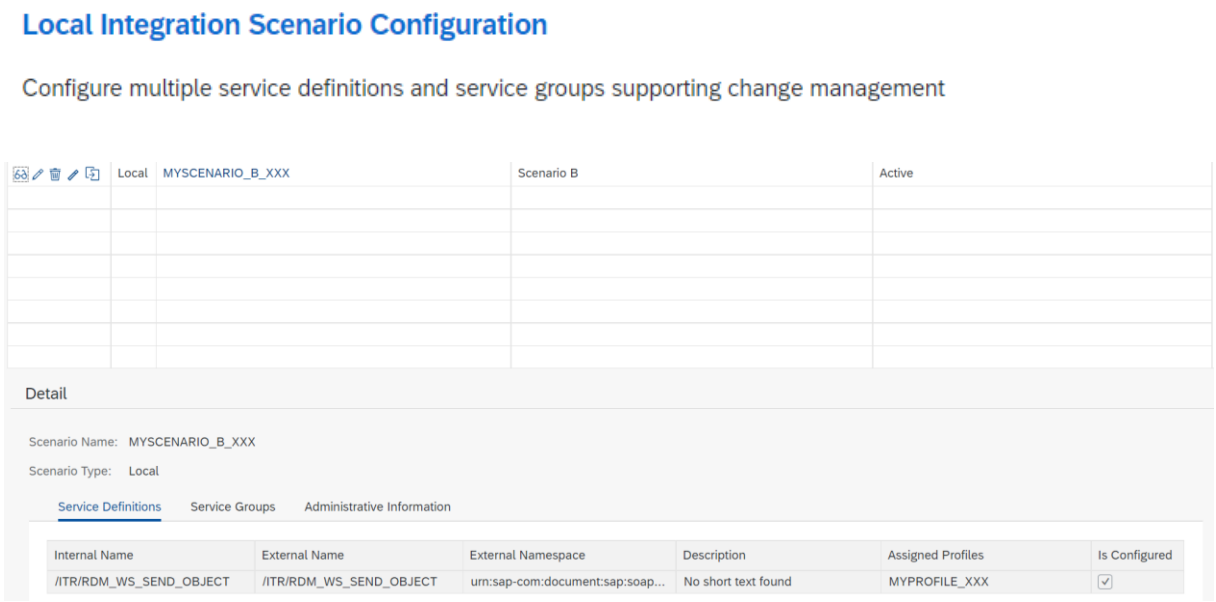
System IR1 200: transaction SOAMANAGER:





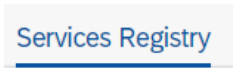
4.2.3 Receiver: Service Administration: Local Integration Scenario

System IR1 200: transaction SOAMANAGER:



4.2.4 Receiver: Services Registry: Published Systems and Objects

System IR1 200: transaction SOAMANAGER:



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 ir1
	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 ir1
	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 ir1

Details of Service Definition: /ITR/RDM_WS_SEND_OBJECT

Overview | Configurations | Classifications | Details

Define Services and Bindings

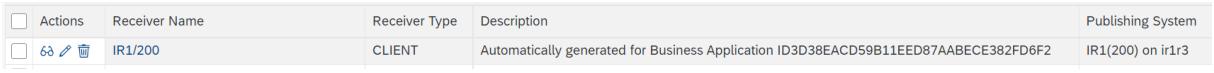
Create Service | Activate | Deactivate | Delete | Republish | Display as List

Service/Binding	Actions	State	Description
B330F036C1921EDD87C2B0AEB7C14D2B		Active	
BINDING__-ITR_-RDM_WS_SEND_OBJECT_MYPROFILE_XXX_L			
B330F036C1921EED87C87B02F24F87E6		Active	
BINDING__-ITR_-RDM_WS_SEND_OBJECT_MYPROFILE_XXX_L			
Z_IR1200_WS_SEND_OBJECT		Active	Z_IR1200_WS_SEND_OBJECT
Z_IR1200_WS_SEND_OBJECT			

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

Published IBCs

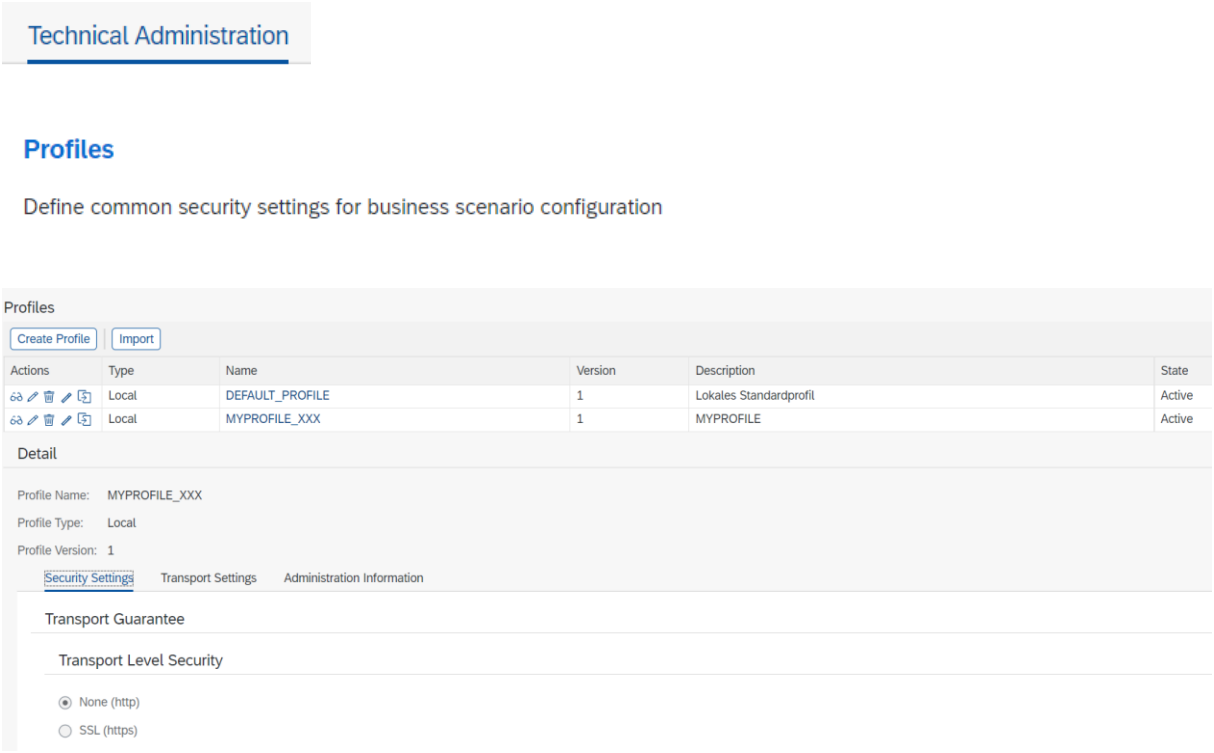
Display and maintain published Identifiable Business Contexts (IBCs) in Services Registry



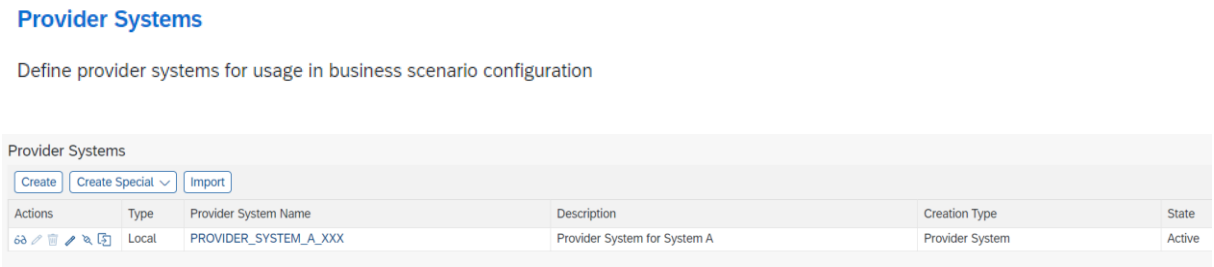
Actions	Receiver Name	Receiver Type	Description	Publishing System
	IR1/200	CLIENT	Automatically generated for Business Application ID3D38EACD59B11EED87AABECE382FD6F2	IR1(200) on ir1r3

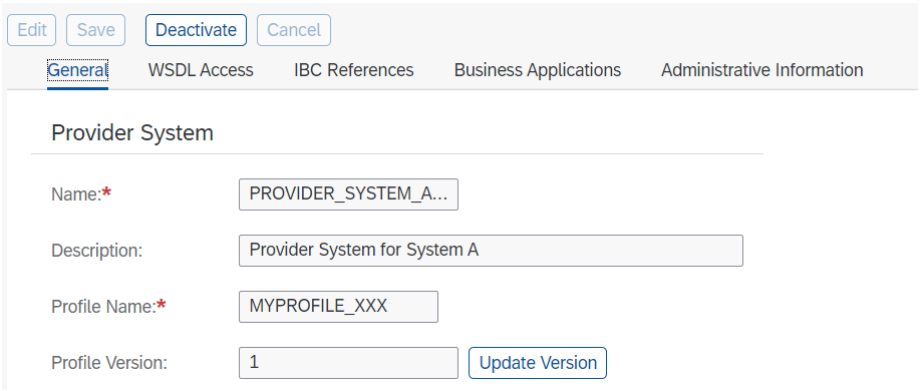
4.2.5 Sender: Technical Administration: Profiles and Provider Systems

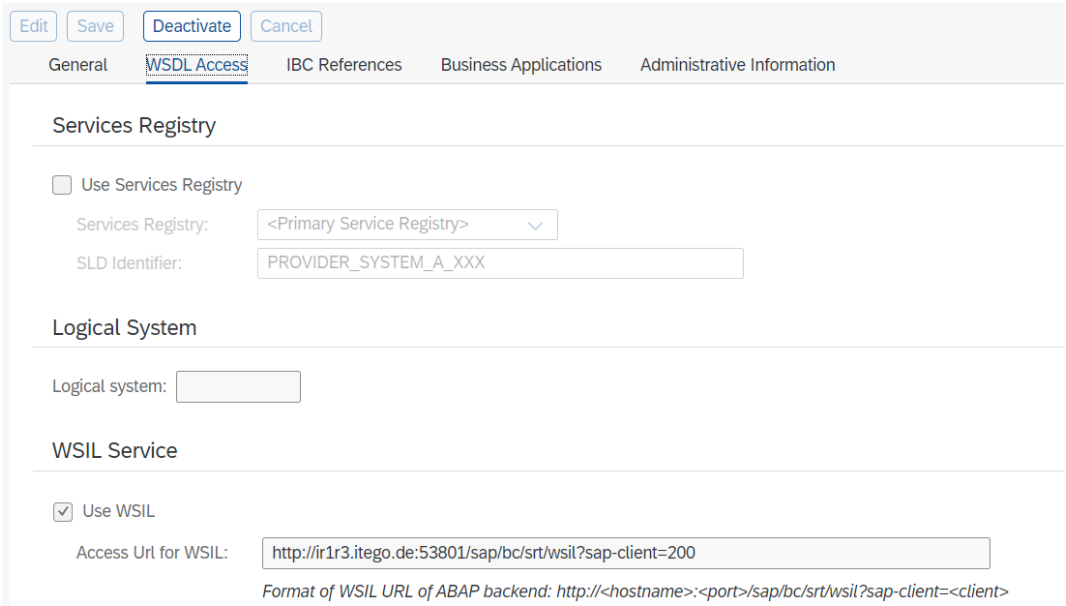
System IH1 100: transaction SOAMANAGER:

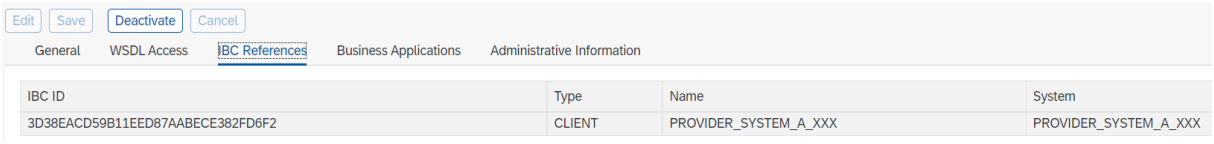


Or https based on your security requirements.



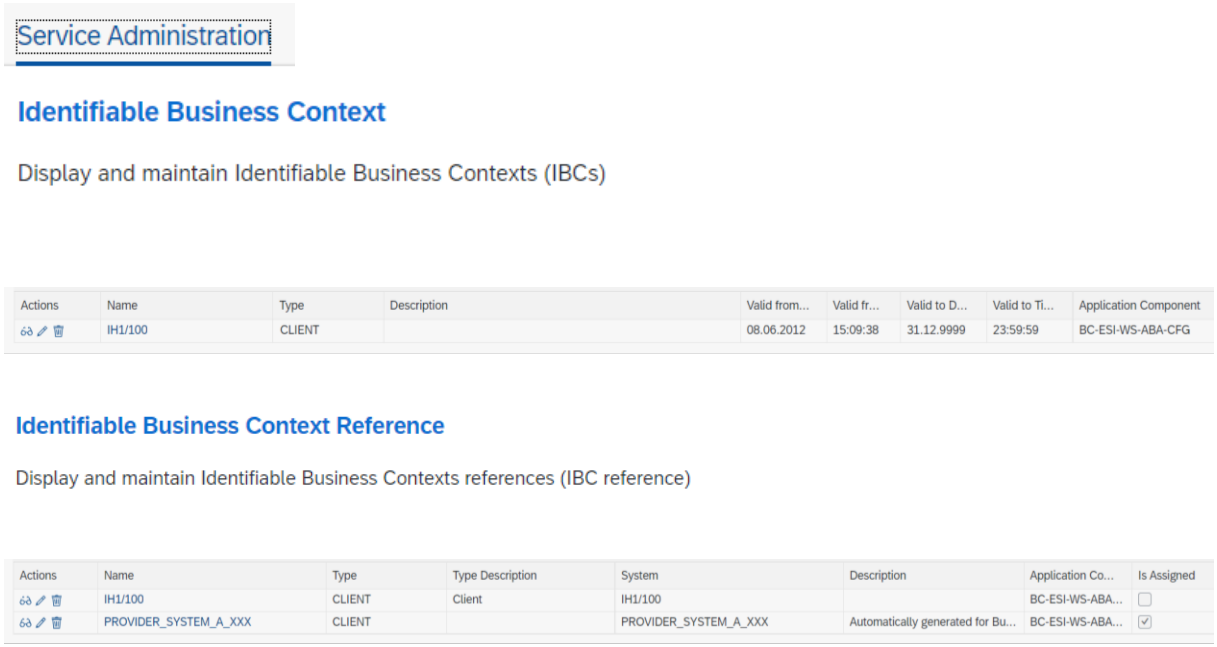






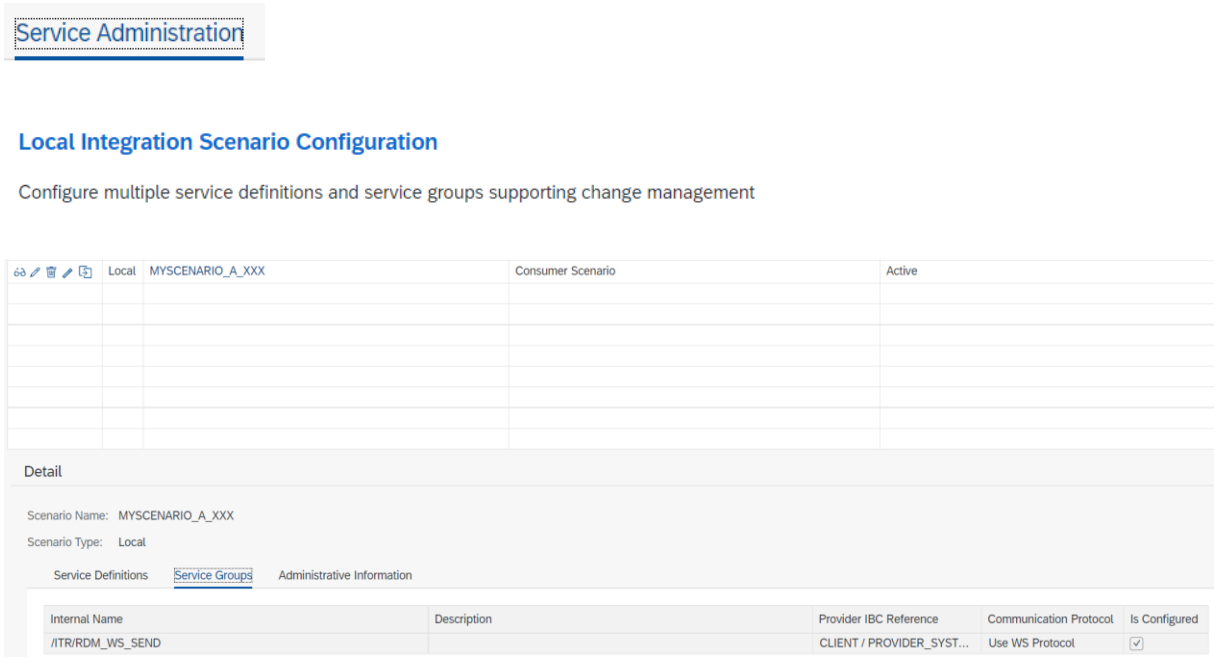
4.2.6 Sender: Service Administration: Business Context

System IH1 100: transaction SOAMANAGER:



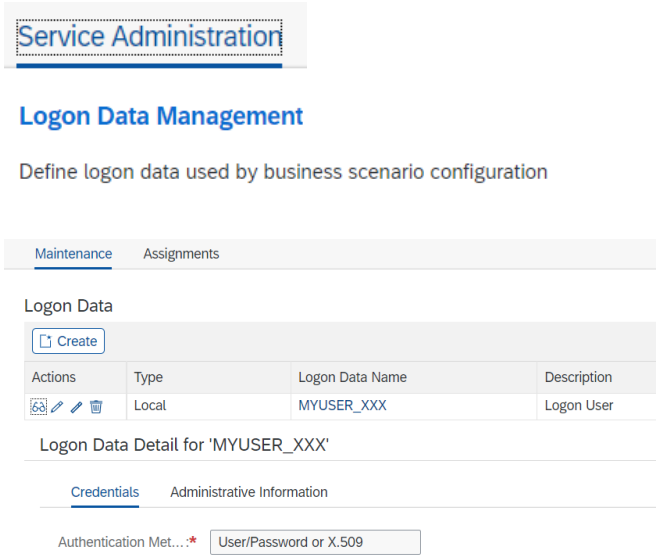
4.2.7 Sender: Service Administration: Local Integration Scenario

System IH1 100: transaction SOAMANAGER:

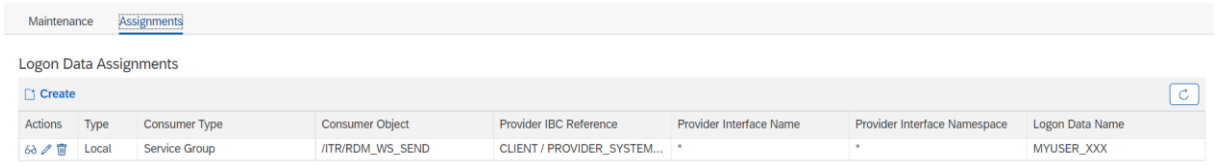


4.2.8 Sender: Service Administration: Logon Data

System IH1 100: transaction SOAMANAGER:



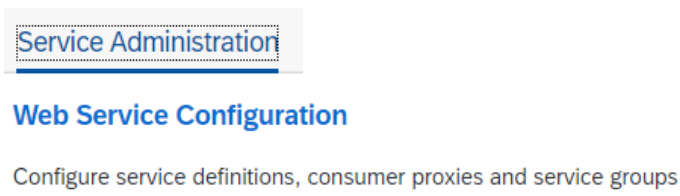
Provide User and Password

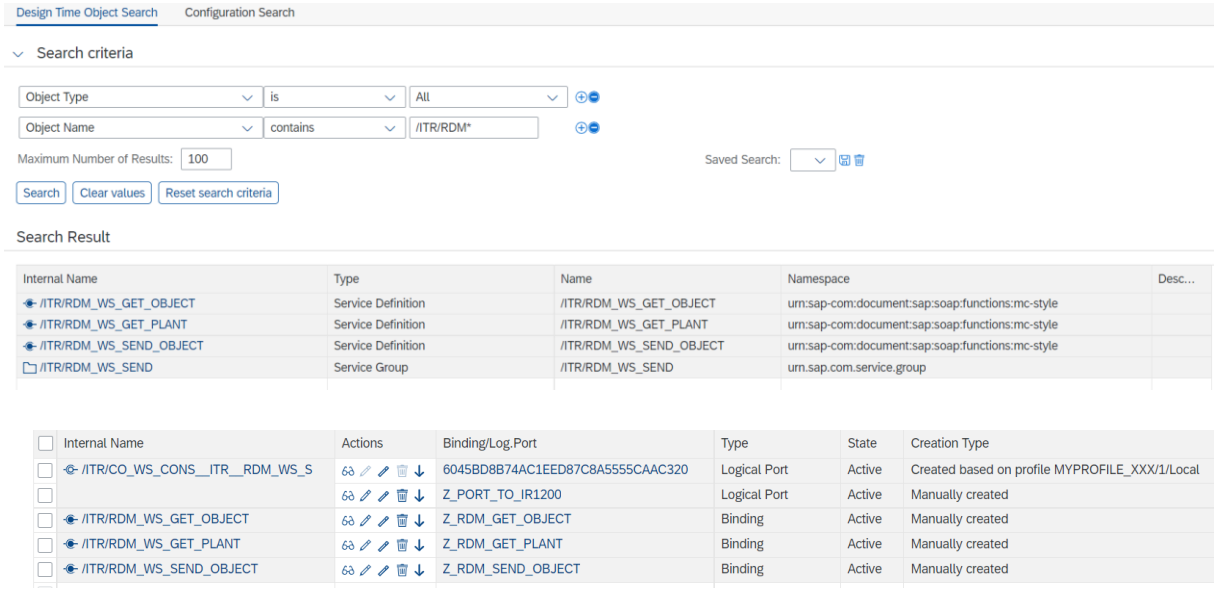


Assignment to Service Group

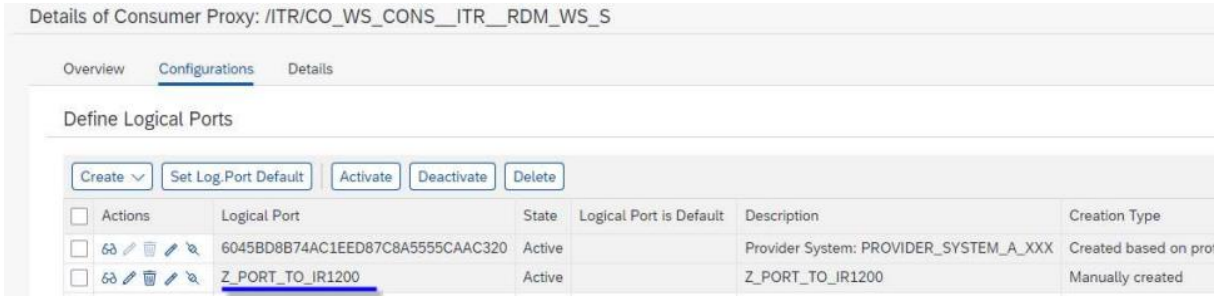
4.2.9 Sender: Service Administration: Web Service

System IH1 100: transaction SOAMANAGER:



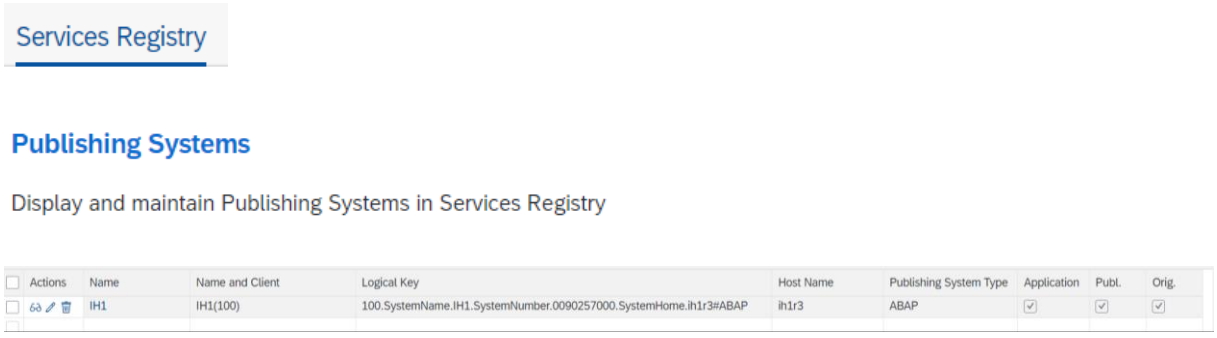


Logical port created based on exported WSDL from receiver system.



4.2.10 Sender: Services Registry: Published Systems and Objects

System IH1 100: transaction SOAMANAGER:



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_GET_OBJECT	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_GET_OBJECT	Configured		IH1(100) on ih1r3
	/ITR/RDM_WS_GET_PLANT	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_GET_PLANT	Configured		IH1(100) on ih1r3
	/ITR/RDM_WS_SEND_OBJECT	urn:sap-com:document:sap:soap:functions:mc-style	/ITR/RDM_WS_SEND_OBJECT	Configured		IH1(100) on ih1r3

Published Bindings

Display and maintain published Bindings in Services Registry

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