Master Data Governance for Customer
Caution
Before you start the implementation, make sure you have the latest version of this document. You can find the latest version at the following location: xxx /xxx

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

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2015-06-03</td>
<td>Final Version</td>
</tr>
</tbody>
</table>
## Content

1. **Master Data Governance for Customer** ............................................ 5

2. **Services to be Activated for MDG Web Dynpro Applications** ................. 6

3. **Configuring Master Data Governance for Customer** ................................. 13
   3.1 Activation of the Business Partner (BP) Data Model .............................. 16
   3.2 Activate Business Configuration Set .................................................. 17
   3.3 Create Number Range Interval for Cleansing Application ....................... 17
   3.4 Define Number Ranges for Customer Account Groups and Contact Persons per Target System ................................................................. 18
   3.5 Configure Change Request Settings .................................................... 18
   3.6 Specify Change Request Types to Enable Account Group Change ............... 26
   3.7 Set Up the Workflow ............................................................................ 27
   3.8 Set Up the Rule-Based Workflow .......................................................... 31
   3.9 Show Master Data Governance for Customer 6.1 Specific UIBBs ................ 35
   3.10 Show Master Data Governance for Customer 9.0 Specific UIBBs ............... 37
   3.11 Set Up Business Partner ...................................................................... 39
   3.12 Set Up Customer .................................................................................. 40
   3.13 Set Up Customer-Vendor Integration .................................................... 40
   3.14 Replicate Data ..................................................................................... 40
   3.15 Define Value Mapping .......................................................................... 40
   3.16 Define Key Mapping ............................................................................ 41
   3.17 Search and Duplicate Check .................................................................. 41
   3.18 Data Quality Services .......................................................................... 41
   3.19 Set Up Embedded Search ...................................................................... 41
   3.20 Configuring SAP HANA-Based Search for MDG .................................... 42
   3.21 Choose the UI environment you want to use to run SAP Master Data Governance ......................................................................................... 48
   3.22 Data Transfer of Customer Master Data .................................................. 50
   3.23 Event Control ....................................................................................... 52
   3.24 Validations and Enrichments .................................................................. 53
   3.25 Set Up Address Screening ...................................................................... 53

4. **Configuring Master Data Governance for Customer on Client System** ........ 55

5. **Adapting Master Data Governance for Customer** ..................................... 59
   5.1 Data Modeling ....................................................................................... 59
      - Concept of the MDG Data Modeling ..................................................... 59
      - Transportation of Data Models to the Target System ............................ 64
      - Defining Data Models in the Configuration Workbench ......................... 65
   5.2 UI Modeling ......................................................................................... 68
      - Managing of UI Configurations ............................................................ 68
      - Context-Based Adaptation for Business Partner ................................... 70
      - Configuring Business Context Viewer for MDG Customer ................... 70
   5.3 Data Quality and Search ........................................................................ 71
<table>
<thead>
<tr>
<th>Search Providers for Master Data Governance</th>
<th>71</th>
</tr>
</thead>
<tbody>
<tr>
<td>Definition of Validations and Derivations in BRFplus</td>
<td>82</td>
</tr>
<tr>
<td>Data Quality Remediation Configuration Guide</td>
<td>87</td>
</tr>
<tr>
<td>5.4 Process Modeling</td>
<td>89</td>
</tr>
<tr>
<td>Defining a Governance Scope</td>
<td>89</td>
</tr>
<tr>
<td>Configuration of the Change Request Process</td>
<td>90</td>
</tr>
<tr>
<td>Scope for Hierarchy-Specific Changes</td>
<td>111</td>
</tr>
<tr>
<td>Navigation to the User Interface</td>
<td>114</td>
</tr>
<tr>
<td>Enabling Detailed Analysis of Change Requests</td>
<td>118</td>
</tr>
<tr>
<td>5.5 Data Replication</td>
<td>122</td>
</tr>
<tr>
<td>Configuring Data Replication</td>
<td>123</td>
</tr>
<tr>
<td>Data Replication Using Enterprise Service Oriented Architecture (SOA)</td>
<td>131</td>
</tr>
<tr>
<td>Data Replication of Customer Master Data Using ALE</td>
<td>132</td>
</tr>
<tr>
<td>Data Replication of Business Partner Master Data Using ALE</td>
<td>136</td>
</tr>
<tr>
<td>Configuration for Data Replication to CRM Systems</td>
<td>140</td>
</tr>
<tr>
<td>5.6 Value Mapping</td>
<td>145</td>
</tr>
<tr>
<td>Value Mapping: Use of Global Code Lists</td>
<td>145</td>
</tr>
<tr>
<td>Value Mapping: Use of System-Specific Code Lists</td>
<td>147</td>
</tr>
<tr>
<td>5.7 Key Mapping</td>
<td>148</td>
</tr>
<tr>
<td>5.8 Governance Application Programming Interface</td>
<td>148</td>
</tr>
<tr>
<td>6 Time Dependency</td>
<td>151</td>
</tr>
<tr>
<td>7 Interlocking</td>
<td>152</td>
</tr>
</tbody>
</table>
1 Master Data Governance for Customer

Master Data Governance for Customer enables you to monitor and control the creation, editing, and deletion of business partner master data.

This documentation provides the information you require to set up Master Data Governance for Customer. It supplements the information provided in Customizing as well as the information about activities that you need to execute in addition to configuring Customizing settings.
2 Services to be Activated for MDG Web Dynpro Applications

For security reasons the services delivered for Web Dynpro applications initially are available in an inactive state only. You have to activate the services you want to use.

Activities

To activate the services, proceed as described below:

1. On the Maintain Services screen (transaction SICF), make sure that the Hierarchy Type SERVICE is selected, enter the Service Name, and choose Execute.
2. Choose Service/Host Activate to activate the service.

Note

You have to perform the procedure for each single service you want to activate. Once you have activated a service it cannot be reset to inactive.

The table below provides a list of the services used in the respective components of SAP MDG, central governance.

Table 2

<table>
<thead>
<tr>
<th>Service</th>
<th>Name</th>
<th>MDG-C / MDG-S / MDG-BP</th>
<th>MDG-M</th>
<th>MDG-F</th>
<th>MDG-CO</th>
</tr>
</thead>
<tbody>
<tr>
<td>APB_LAUNCHPAD</td>
<td>Launchpad</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>BS_OVP_BP</td>
<td>Web Dynpro Component for BP OVP</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BS_OVP_BP_MRP</td>
<td>Business Partner Multi Objects Processing Application</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BS_OVP_CC</td>
<td>Cleansing Case Application</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONFIGURE_APPLICATION</td>
<td>Application Configuration</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>CONFIGURE_COMPONENT</td>
<td>Configure Component</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>CUSTOMIZE_COMPONENT</td>
<td>Component Configurator for</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Service</td>
<td>Name</td>
<td>MDG-C / MDG-S / MDG-BP</td>
<td>MDG-M</td>
<td>MDG-F</td>
<td>MDG-CO</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------</td>
<td>-------------------------</td>
<td>-------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>DRF_ADHOC_REPLICATION</td>
<td>Adhoc Replication Model</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>DRF_FILTER_BO_FPM</td>
<td>Filter Criteria</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>DRF_FILTER_POWL_AC</td>
<td>Application Configuration for Filter POWL</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>DRF_FILTER_POWL_QAF_AC</td>
<td>Filter Maintenance POWL</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>DRF_FPM_OIF_MONITORING</td>
<td>Monitoring Web Dynpro Application</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>DRF_FPM_SEG_FLTR_POPUP_AC</td>
<td>Application configuration for the popup</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>DRF_MANUAL_REPLICATION</td>
<td>Manual Replication</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>FPM_CFG_HIERARCHY_BROWSER</td>
<td>FPM Application Hierarchy Browser</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>IBO_WDA_INBOX</td>
<td>Lean Workflow Inbox Application</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_ANLY_CR_REJ_REASON</td>
<td>Change Request Rejection reason</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>MDG_BS_CONVERTOR</td>
<td>Master Data File Convertor</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDG_BS_DATALOAD_MONITOR</td>
<td>Reprocessing</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_BS_DLDISPLAY_LOG</td>
<td>Web Dynpro Application MDG_BS_DL_DISPLAY_LOG</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_BS_DL_MONITOR_CONF</td>
<td>Data Load Monitor</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_BS_FILE_IMPORT</td>
<td>Application for File Import</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

Master Data Governance for Customer Services to be Activated for MDG Web Dynpro Applications
<table>
<thead>
<tr>
<th>Service</th>
<th>Name</th>
<th>MDG-C / MDG-S / MDG-BP</th>
<th>MDG-M</th>
<th>MDG-F</th>
<th>MDG-CO</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDG_BS_GEN_MC_OVP</td>
<td>Generic Mass Change Application</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>MDG_BS_MAT</td>
<td>MDG-M: UI (entry point)</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDG_BS_MAT_MC</td>
<td>MDG-M: Mass Change UI</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>MDG_BS_MAT_OVP</td>
<td>MDG-M: UI with CBA</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>MDG_BS_MAT_SE_ARCH</td>
<td>MDG-M: UI, Search</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>MDG_BS_WD_ANALYSE_IDM</td>
<td>Analyse ID Web Dynpro</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_BS_WD_ID_MATCH_SERVICE</td>
<td>Web Dynpro Application MDG_BS_WD_ID_MATCH_SERVICE</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_BS_WD_RSI_DISPLAY</td>
<td>Display Replication Status Display</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_CREQUEST_GRAPH_ANALYSIS</td>
<td>Application for Flash</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>MDG_CR_PROCESS_TTIME_TREE</td>
<td>Processing Time</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>MDG_DATALOAD_EXPORT_WDA</td>
<td>Export Master Data and Mapping Information</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDG_DISPLAY_COLORS</td>
<td>Cell Colors used for Highlighting Changes</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_DQR_OVP</td>
<td>OVP for MDG Data Quality Remediation</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>MDG_EXTR_FPM_CMP</td>
<td>Extractor</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>MDG_FILE_UPLOAD_CMP</td>
<td>File Uploader</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service</td>
<td>Name</td>
<td>MDG-C / MDG-S / MDG-BP</td>
<td>MDG-M</td>
<td>MDG-F</td>
<td>MDG-CO</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
<td>------------------------</td>
<td>-------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>MDG_MONITOR_C R_PROCESS_TIME</td>
<td>Application Configuration for Monitoring CR Processing Time</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>MDG_TRANSFORM ER_FPM_CMP</td>
<td>Transformer component for FPM</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>MDGF_OVP_GEN</td>
<td>MDG-F Application</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>OIF_CFG_CENTER</td>
<td>BCV Configuration Center (FPM)</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>POWL</td>
<td>Personal Object Work List</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>USMD_APPLICATION_LOG</td>
<td>Web Dynpro Application USMD_APPLICATION_LOG</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_BRFPLUS_CATALOG_BROWSER</td>
<td>BRFplus Catalog Browser</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_CHANGE_DOCUMENT</td>
<td>Change Documents</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_CREQUEST_CREATE</td>
<td>Create Change Request</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_CREQUEST_PROCESS</td>
<td>USMD_CREQUEST_PROCESS</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_CREQUEST_PROTOCOL2</td>
<td>Workflow Information</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_DISTRIBUT E</td>
<td>Web Dynpro Application USMD_DISTRIBUT E / Component FPM_OIF_COMPONENT</td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_EDITION</td>
<td>Edition</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>USMD_EDITION_COMPARE</td>
<td>Edition Comparison</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>USMD_EDITION_C REQUEST</td>
<td>Display of Change Requests of an Edition</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Service</td>
<td>Name</td>
<td>MDG-C / MDG-S / MDG-BP</td>
<td>MDG-M</td>
<td>MDG-F</td>
<td>MDG-CO</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------</td>
<td>------------------------</td>
<td>-------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>USMD_EDITION_HISTORY2</td>
<td>Edition History</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_ENTITY</td>
<td>Collective Processing of an Entity</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_ENTITY_SEARCH</td>
<td>Search for Entities</td>
<td></td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_ENTITY_VALUE2</td>
<td>Single Processing of an Entity</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_FILE_DOWNLOAD</td>
<td>File Download</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_FILE_UPLOAD</td>
<td>File Upload</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_ISR_PROCESS</td>
<td>ISR Processing of a Change Request</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_MASS_CHANGE</td>
<td>Mass Change</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_OVPGENERATION</td>
<td>MDG: Application for Custom Objects</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_REMOTE.WhereUsed</td>
<td>Remote Where-Used List</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_RULE</td>
<td>Rule Engine Configuration for Validation and Derivation</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_SEARCH</td>
<td>MDG Generic Search</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_UI_CONFIGURATION</td>
<td>Manage UI Configuration</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>USMD_SSW_RULE</td>
<td>Definition of Rules for Rule-Based Workflow</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_WF_NAVIGATION</td>
<td>Workflow-Based Navigation</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>USMD_WHEREUSED</td>
<td>Where-Used List</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Service</td>
<td>Name</td>
<td>MDG-C / MDG-S / MDG-BP</td>
<td>MDG-M</td>
<td>MDG-F</td>
<td>MDG-CO</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>------------------------</td>
<td>-------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>WDA_AUTH_OIF_ACL_FRAME</td>
<td>ACL Maintenance</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_BS_ANLY_LIST</td>
<td>Simplified Reporting: Simple List on BI Query</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_BS_ANLY_LIST_OVP</td>
<td>List</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_CFG_ENTRY</td>
<td>Entry Sheet of BCV Configuration Center (POWL)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_CFG_GAF_WIZARD</td>
<td>Configuration Wizard</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_CFG_LAUNCHPAD</td>
<td>Launchpad Maintenance</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_CFG_OIF_UGRP</td>
<td>Web Dynpro Application /BCV/WDA_CFG_OIF_UGRP / Component FPM_OIF_COMPO</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_MDG_DT_CONF_WORKBENCH</td>
<td>Configuration Workbench</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_OIF_MANAGE</td>
<td>Manage Interface Models</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_OIF_DISPLAY</td>
<td>Display OIF Model</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_OIF_CREATE</td>
<td>Create Outbound Interface</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_OIF.WhereUsed</td>
<td>Interface Models Usage</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_QRM_BRF_OBJMAN</td>
<td>BRFplus Object Manager</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_SMT</td>
<td>Service Mapping Tool Web Dynpro Application</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_UIF_DASHBOARD</td>
<td>PCV Dashboard</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDA_UIF_MAIN</td>
<td>PCV Main</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Service</td>
<td>Name</td>
<td>MDG-C / MDG-S / MDG-BP</td>
<td>MDG-M</td>
<td>MDG-F</td>
<td>MDG-CO</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------</td>
<td>-------------------------</td>
<td>-------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>WDA_UIF_SIDEPA NEL</td>
<td>BCV Side Panel for Standalone Mode</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDC_CFG_PAGE_ BUILDER</td>
<td>Page Builder</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDC_CFG_XC_M ETA</td>
<td>Xcelsius Metadata Extraction Standalone Application</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDC_UIF_CHIP</td>
<td>BCV Chip</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDC_UIF_COCKPI T</td>
<td>BCV Cockpit Start</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WDR_CHIP_PAGE</td>
<td>wdr_chip_page</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WD_GLOBAL_SET TING</td>
<td>Cross-Application Settings for Web Dynpro ABAP</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>WEBGUI</td>
<td>SAP GUI for HTML</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>
3 Configuring Master Data Governance for Customer

SAP Master Data Governance for Customer (MDG-C) enables you to govern customer master data on a hub system and to replicate the customer master data to a variety of different client systems.

Additional features for decentralized customer master data governance are delivered with the business function *Master Data Governance for Customer on Client (ERP)*. It enables you to search and compare corresponding data on the hub system, before creating or changing customer master data on a client system.

**Note**
For detailed information how to configure MDG-C on client system, see *Configuring Master Data Governance for Customer on Client System* [page 55].

To use SAP Master Data Governance for Customer, you need to carry out the steps described below.

**Prerequisites**

**Business Functions**

Before you activate the 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 creating a copy of this role and assigning the relevant authorization values. For the authorization object USMD_DM Data Model you need to assign the values for the authorization field USMD_MODEL Data Model and the values for the authorization activity ACTVT Activity (for example 01: Create or generate, or 02: Change).

In the Customizing activity *Activate Business Functions* (transaction SFW5), you have activated the following business functions:

**MDG-C on a hub system**

- *Master Data Governance, Generic Functions (MDG_FOUNDATION)* [external document]
- *Master Data Governance, Generic Functions 2 (MDG_FOUNDATION_2)* [external document]
- *Master Data Governance, Generic Functions 3 (MDG_FOUNDATION_3)* [external document]
- *Master Data Governance, Generic Functions 7.0 (MDG_FOUNDATION_4)* [external document]
- *Master Data Governance, Generic Functions 7.0 Feature Pack (MDG_FOUNDATION_5)* [external document]

**Note**
This business function is available with support package 02 and higher of MDG 7.0.

- *Master Data Governance, Generic Functions 8.0 (MDG_FOUNDATION_6)* [external document]
- *Master Data Governance, Generic Functions 9.0 (MDG_FOUNDATION_7)* [external document]
- *Master Data Governance for Customer (MDG_ERP_CUSTOMER_1)* [external document]
- *Master Data Governance for Customer 6.1 (MDG_ERP_CUSTOMER_2)* [external document]
• Master Data Governance for Customer 7.0 (MDG_ERP_CUSTOMER_3) [external document]
• Master Data Governance for Customer 7.0 Feature Pack (MDG_ERP_CUSTOMER_4) [external document]

Note
This business function is available with support package 02 and higher of MDG 7.0.

• Master Data Governance for Customer 8.0 Highlight Deletions (MDG_ERP_CUSTOMER_HD) [external document]
• Master Data Governance for Customer 9.0 (MDG_ERP_CUSTOMER_5) [external document]

MDG-C on a client system
• Master Data Governance for Customer on Client (ERP) (MDG_CUST_ERPCLIENT_1) [external document]

Note
If you intend to distribute data using SOA services, you have to activate the following business functions on the hub system and on the client systems:
• Business Foundation: Services for Supplier (CA_SUPPLIER_SOA) [external document]
• Business Foundation: Services for Business Partner (CA_BP_SOA) [external document]

Note
If you want to use graphical elements in the MDG-C work center or the Business Context Viewer (BCV), we recommend to activate the following business functions:
• FND, Business Context Viewer Main Application (/BCV/MAIN) [external document]
• FND, Business Context Viewer Main Application 2 (/BCV/MAIN_1) [external document]
For information about the Business Context Viewer, see Business Context Viewer (BCV) [external document].

Note
If you want to use the ALE replication of customer master records with bank accounts that were created with IBAN and without BIC, you have to activate the following business function on the MDG hub and on the MDG client system:
• Materials Management: Procure to Pay (LOG_MM_CI_2) [external document]

Authorization Objects
You have assigned the relevant authorization objects and roles. For more information about the authorization objects and roles, see Customer Master Data Governance (CA-MDG-APP-CUS) [external document].

Set Up Workflow
You have defined general settings for SAP Business Workflow [external document] in Customizing for SAP NetWeaver under ➤ Application Server ➤ Business Management ➤ SAP Business Workflow ➤

Web Dynpro Applications
You have activated the services for Web Dynpro Applications. For a detailed list of the relevant services, see Services to be Activated for Web Dynpro Applications [page 6].
Constraints

- With MDG being deployed on an ERP system that is not directly connected to a CRM system by CRM Middleware, services are required to distribute business partner and customer data into a CRM client system as of CRM 7.01 and higher.

  Up to CRM 7.01 the distribution is limited to business partner data; as of CRM 7.02 most customer and CRM specific data such as sales area data or tax classifications can be distributed.

Process

You run the settings for this process in Customizing under Cross-Application Components > Processes and Tools for Enterprise Applications > Master Data Governance, Central Governance.

Note

You can access all MDG specific Customizing using the transaction MDIMG.

1. Activation of the Business Partner (BP) Data Model [page 16]
2. Activate Business Configuration Set [page 17]
3. Create Number Range Interval for Cleansing Application [page 17]
4. Define Number Ranges for Customer Account Groups and Contact Per [page 18]
5. Configure Change Request Settings [page 18]
6. Specify Change Request Types to Enable Account Group Change [page 26]
7. Set Up the Workflow [page 27]
8. Set Up the Rule-Based Workflow [page 31]
9. Show Master Data Governance for Customer 6.1 Specific UIBBs [page 35]
10. Show Master Data Governance for Customer 9.0 Specific UIBBs [page 37]
11. Set Up Business Partner [page 39]
12. Set Up Customer [page 40]
13. Set Up Customer-Vendor Integration [page 40]
14. Replicate Data [page 40]
15. Define Value Mapping [page 40]
16. Define Key Mapping [page 41]
17. Search and Duplicate Check [page 41]
18. Data Quality Services [page 41]
19. Set Up Embedded Search [page 41]
20. Configuring SAP HANA-Based Search for MDG [page 42]
21. Choose the UI environment you want to use to run SAP Master Data [page 48]
22. Data Transfer of Customer Master Data [page 50]
23. Event Control [page 52]
24. Validations and Enrichments [page 53]
25. Set Up Address Screening [page 53]
Result

The system is configured for Master Data Governance for Customer.

More Information

Master Data Governance Security Guide [external document]
For information about the data model of Master Data Governance for Customer and options to extend the data model, see SAP Note 1973686.

3.1 Activation of the Business Partner (BP) Data Model

**Note**
When you activate the MDG_ERP_CUSTOMER_3 business function the system activates the BP data model. Before continuing with system configuration, open transaction SFW5 and check if the activation has been completed by choosing Goto Switch Framework Logs and selecting the corresponding entry with a double click.

1. Check whether you can use the data model delivered by SAP for managing your customer master data with MDG-C.
   If you want to enhance the delivered data model, edit it in the Customizing for Master Data Governance, Central Governance under General Settings Data Modeling Edit Data Model and activate it again.

   **Note**
   You can select the governance scope at any point after you activate the BP data model. The governance scope determines which fields can be edited and which fields are read-only on the MDG-C UI. To define the governance scope run this Customizing activity: Master Data Governance, Central Governance General Settings Process Modeling Define Governance Scope.
   To restore the BP data model to its state in enhancement package 6 for SAP ERP 6.0 run this Customizing activity: Master Data Governance, Central Governance Master Data Governance for Business Partner Activate BC Set to Reset Governance Scope.
   For further information, see Defining a Governance Scope [external document].

2. Assign an internal key
   To support internal key assignment, run the activity in Customizing for Master Data Governance, Central Governance under General Settings Data Modeling Define Prefixes for Internal Key Assignment.
   **Example:** If you create new entries with the following values, all temporary IDs are prefixed with $ and an internal sequence number.
   Data Model = BP, Entity Type = ADDRNO => Prefix $(where ADDRNO is the address number)
   Data Model = BP, Entity Type = BP_HEADER => Prefix $(where BP_HEADER is the customer number)
### 3.2 Activate Business Configuration Set

If you intend to use the Customer UI, run the corresponding activity in Customizing for **Master Data Governance, Central Governance** under 🔄 Master Data Governance for Customer ➔ Import Predefined Change Request Types.

**Note**

MDG-C offers two different groups of change request types that support either the Customer UI or the ERP Customer UI. Only one of these groups should be used. If you intend to use the ERP Customer UI as alternative to the Customer UI, proceed as follows:

1. Run the corresponding activity in Customizing for Master Data Governance, Central Governance under 🔄 Master Data Governance for Customer ➔ ERP-Customer UI (Alternative to the Customer UI) ➔ Import Predefined Change Request Types for ERP-Customer UI.

**Note**

To access the BC-Set, open the assigned documentation and choose the link.

2. Proceed with the ERP Customer UI specific settings described in the corresponding note in the document Configure Change Request Settings [page 18].

**Note**

The change request types are delivered with the following BC sets with the business function Master Data Governance for Customer 9.0 [external document]:

- **MDGC Change Request Types (Customer) 9.0** CA-MDG-APP-CUS_VC_USMD110_C05 for the customer UI
- **MDGC: Change Request Types (Customer Like UI) 7.0 FP** CA-MDG-APP-CL_VC_USMD110_C02 for the ERP Customer UI

**Note**

To access the BC-Set open the assigned documentation and choose the link.

**Note**

If you want to use the MDG-C work center or side panel or the Business Context Viewer (BCV), you must activate the BC set MDGAP_BCV under 🔄 SAP Menu ➔ Tools ➔ Customizing ➔ Business Configuration Sets ➔ Activation of BC Sets.

### 3.3 Create Number Range Interval for Cleansing Application

If you intend to use the business partner cleansing application carry out the following steps in your MDG hub system.

1. Run transaction COM_CLEAR_NUM, choose 🖼 (Display Intervals) and make sure that the number range interval 01 is available.

2. If the number range interval 01 is not available, run transaction COM_CLEAR_NUM, choose 🖼 (Change Intervals) and create interval 01. For example:
3.4 Define Number Ranges for Customer Account Groups and Contact Persons per Target System

If you intend to override the standard settings, so that you can distribute customer master data into the target systems, with the numbers of the customers in the target system being different from the corresponding numbers in the hub system, run the corresponding activities in Customizing:

- **Account Groups:** Master Data Governance, Central Governance under Master Data Governance for Customer > Define Number Ranges for Customer Account Groups per Target System
- **Contact Persons:** Master Data Governance, Central Governance under Master Data Governance for Customer > Define Number Ranges for Customer Contact Persons per Target System

3.5 Configure Change Request Settings

Depending on your company’s requirements, you might want to adjust and enhance the change request default values.

The following information describes exemplary settings required for a standard governance process. For more information about each Customizing activity, see the relevant documentation for that Customizing activity.

1. Review and/or define which statuses the change requests can have, and which processing options are enabled for those statuses. Optionally, you can add new statuses to be used in the change request types defined in the next step.

   For more information, see Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Change Requests > Edit Statuses of Change Requests.

   The following statuses are required for the SAP standard process:

<table>
<thead>
<tr>
<th>Status Value</th>
<th>Description</th>
<th>Permitted Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>To Be Evaluated</td>
<td>Change of Object List</td>
</tr>
<tr>
<td>01</td>
<td>To Be Considered and Approved</td>
<td>Change of Object List</td>
</tr>
<tr>
<td>02</td>
<td>Changes to Be Executed</td>
<td>Execution of Changes</td>
</tr>
<tr>
<td>03</td>
<td>To Be Revised</td>
<td>Change of Object List</td>
</tr>
<tr>
<td>04</td>
<td>Final Check to Be Performed</td>
<td>No Processing</td>
</tr>
<tr>
<td>Status Value</td>
<td>Description</td>
<td>Permitted Processing</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>05</td>
<td>Final Check Approved</td>
<td>No Processing</td>
</tr>
<tr>
<td>06</td>
<td>Final Check Rejected</td>
<td>No Processing</td>
</tr>
<tr>
<td>07</td>
<td>Activation Failed</td>
<td>No Processing</td>
</tr>
<tr>
<td>08</td>
<td>Approved, to be Replicated</td>
<td>No Processing</td>
</tr>
<tr>
<td>09</td>
<td>Dependent Data to Be Processed/Approved</td>
<td>Execution of Changes</td>
</tr>
<tr>
<td>10</td>
<td>To Revise: Perform Changes</td>
<td>Execution of Changes</td>
</tr>
<tr>
<td>11</td>
<td>Process Errors After Activation</td>
<td>Execution of Changes</td>
</tr>
<tr>
<td>12</td>
<td>Approved, Contact Person to be Processed</td>
<td>No Processing</td>
</tr>
<tr>
<td>99</td>
<td>No Status Set</td>
<td>No Processing</td>
</tr>
</tbody>
</table>

2. Check in the Customizing that the following business activities are in your system under Master Data Governance, Central Governance > General Settings > Process Modeling > Business Activities > Create Business Activity and make sure that they are assigned to the default data model BP.
   - CUP1 (Create Customer)
   - CUP2 (Process Customer)
   - CUP3 (Display Customer)
   - CUP5 (Block/Unblock Customer)
   - CUP6 (Mark Customer for Deletion)
   - CUPM (Multi-Object Processing for Customer)
   - CUF1 (Create Customer (SAP Fiori))
   - BPPI (MDG Hub Inbound: Create Customer)
   - BPPC (Business Partner Data Cleansing)
   - BPPU (MDG Hub Inbound: Process Customer)
   - BPPH (Process Business Partner Hierarchies)
   - BPPL (Business Partner Initial Load)
   - BPPM (Business Partner Mass Maintenance)
   - BPF1 (Create Business Partner (SAP Fiori))

3. Create new change request types for data model BP, or validate after import using business configuration set (BC Set).

For more information, see Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Change Requests > Create Change Request Type.

1. The following table shows the change request types for data model BP of the customer UI. Only the relevant columns are included.
<table>
<thead>
<tr>
<th>Change Request Type</th>
<th>Data Model</th>
<th>Description</th>
<th>Single Object</th>
<th>Main Entity Type</th>
<th>Workflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUST1P2</td>
<td>BP</td>
<td>Create Customer (parallel WF)</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS54400001</td>
</tr>
<tr>
<td>CUST2P1</td>
<td>BP</td>
<td>Process Customer</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS54300003</td>
</tr>
<tr>
<td>CUST5P1</td>
<td>BP</td>
<td>Block/Unblock Customer</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS54300003</td>
</tr>
<tr>
<td>CUST6P1</td>
<td>BP</td>
<td>Mark Customer for Deletion</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS54300003</td>
</tr>
<tr>
<td>CUSTHI01</td>
<td>BP</td>
<td>Local Maintenance: Create Customer</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS46000023</td>
</tr>
<tr>
<td>CUSTHI02</td>
<td>BP</td>
<td>Local Maintenance: Process Customer</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS46000027</td>
</tr>
<tr>
<td>CUSTMRP1</td>
<td>BP</td>
<td>Multi Processing for Customer Financials</td>
<td>No</td>
<td>BP_HEADER</td>
<td>WS54300003</td>
</tr>
<tr>
<td>CUSTMRP2</td>
<td>BP</td>
<td>Multi Processing for Customer Sales</td>
<td>No</td>
<td>BP_HEADER</td>
<td>WS54300003</td>
</tr>
<tr>
<td>BPHP1</td>
<td>BP</td>
<td>Process Business Partner Hierarchies</td>
<td>No</td>
<td>BP_HEADER</td>
<td>WS60800095</td>
</tr>
<tr>
<td>BPLP1</td>
<td>BP</td>
<td>Business Partner Initial Load</td>
<td>No</td>
<td>BP_HEADER</td>
<td>WS72100006</td>
</tr>
<tr>
<td>BPMP1</td>
<td>BP</td>
<td>Business Partner Mass Maintenance</td>
<td>No</td>
<td>BP_HEADER</td>
<td>WS60800095</td>
</tr>
<tr>
<td>BPCC2</td>
<td>BP</td>
<td>Process Business Partner Cleansing Case</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS60800086</td>
</tr>
<tr>
<td>CUST1P3</td>
<td>BP</td>
<td>Create Customer w. Hierarchy Assignment</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS54400003</td>
</tr>
<tr>
<td>CUST2P2</td>
<td>BP</td>
<td>Process Customer w. Hierarchy Assignment</td>
<td>Yes</td>
<td>BP_HEADER</td>
<td>WS54300003</td>
</tr>
</tbody>
</table>
The following settings should exist in the substructures of the change request types:

Substructure of Change Request Types:

**CUST1P2:**

- **Note**
  
  The change request type CUST1P2 supports distributed workflow for customers. It is delivered with the business function *Master Data Governance for Customer 7.0 Feature Pack* [external document].

- **Entity types:**
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard

- **Business Activity:** CUP1

**CUST2P1:**

- **Entity types:**
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard

- **Business Activity:** CUP2

**CUST5P1:**

- **Entity type:** BP_HEADER: Msg. Output = Standard

- **Business Activity:** CUP5

**CUST6P1:**

- **Entity type:** BP_HEADER: Msg. Output = Standard

- **Business Activity:** CUP6

**CUSTHI01:**

- **Entity types:**
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard

- **Business Activity:** BPPI

**CUSTHI02:**

- **Entity types:**
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard

- **Business Activity:** BPPU
CUSTMRP1
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_HRCHY: Msg. Output = Standard
- Business Activity: CUPM

CUSTMRP2
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_HRCHY: Msg. Output = Standard
- Business Activity: CUPM

BPHP1
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_HRCHY: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard
  - BP_SUBHRY: Msg. Output = Standard
- Business Activity: BPPH

BPLP1
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_HRCHY: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard
  - BP_SUBHRY: Msg. Output = Standard
- Business Activity: BPPL

BPMP1
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_HRCHY: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard
  - BP_SUBHRY: Msg. Output = Standard
- Business Activity: BPPM
BPCC2
○ Entity types:
  ○ ADDRNO: Msg. Output = Standard
  ○ BP_HEADER: Msg. Output = Standard
  ○ BP_REL: Msg. Output = Standard
○ Business Activity: BPPC

CUST1P3:
○ Entity types:
  ○ ADDRNO: Msg. Output = Standard
  ○ BP_HEADER: Msg. Output = Standard
  ○ BP_REL: Msg. Output = Standard
○ Business Activity: CUP1

CUST2P2:
○ Entity types:
  ○ ADDRNO: Msg. Output = Standard
  ○ BP_HEADER: Msg. Output = Standard
  ○ BP_REL: Msg. Output = Standard
○ Business Activity: CUP2

Note
In an optional step you can specify Service Level Agreement for all the mentioned change request types.

Note
ERP Customer UI Specific Settings
○ If you use the ERP Customer UI, check that the following business activities are in your system:
  ○ CLP1 (Create Customer)
  ○ CLP2 (Process Customer)
  ○ CLP3 (Display Customer)
  ○ CLP5 (Block/Unblock Customer)
  ○ CLP6 (Mark Customer for Deletion)
  ○ BPPL (Business Partner Initial Load)
  ○ BPPM (Business Partner Mass Maintenance)
○ The following table shows the change request types for data model BP of the ERP Customer UI. Only the relevant columns are included.
The following settings should exist in the substructures of the change request types: Substructure of Change Request Types:

**CUSTL1**

Note

The change request type CUSTL1 supports a lean request UI for ERP customers. It is delivered with the business function Master Data Governance for Customer 7.0 Feature Pack [external document].

- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard
- Business Activity: CLP1

**CUSTL1P2**

Note

The change request type CUSTL1P2 supports distributed workflow for ERP customers. It is delivered with the business function Master Data Governance for Customer 7.0 Feature Pack [external document].
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard
- Business Activity: CLP1

CUSTL2P1
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
- Business Activity: CLP2

CUSTL5P1
- Entity type: BP_HEADER: Msg. Output = Standard
- Business Activity: CLP5

CUSTL6P1
- Entity type: BP_HEADER: Msg. Output = Standard
- Business Activity: CLP6

BPHP1:
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_HRCY: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard
  - BP_SUBHRY: Msg. Output = Standard
- Business Activity: BPPL

BPLP1
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
  - BP_HRCY: Msg. Output = Standard
  - BP_REL: Msg. Output = Standard
  - BP_SUBHRY: Msg. Output = Standard
- Business Activity: BPPL

BPMP1
- Entity types:
  - ADDRNO: Msg. Output = Standard
  - BP_HEADER: Msg. Output = Standard
3.6 Specify Change Request Types to Enable Account Group Change

You can specify change request types to enable the change of account groups. In the corresponding change requests the field *Account Group* will then be changeable instead of being a mere display field.

If you want to enable all users to change account groups, then specify your standard change request types.

If you want to enable only a restricted group of user we recommend to use a specific change request type for this task.

**Note**

Changing the account group might result in different partner functions, different number assignment and different attributes for certain fields.

**Preconditions**

- You have created the corresponding change request types.
- You have assigned the change requests types to the Business Partner (BP) Data Model.
Activity
Run the Customizing activity Specify Change Request Types to Enable Account Group Change under Master Data Governance, Central Governance > Master Data Governance for Customer.

More Information
For further information, see SAP Note 1952436.

3.7 Set Up the Workflow
This section describes how to make the Customizing settings required to run the workflow for the approval process in MDG-C.

You define the workflow settings in Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Workflow.

1. Activate type linkage
To activate the type linkage, run the following activity in Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Workflow > Activate Type Linkage.

Ensure that the object type BUS2250 has the following settings:
- Event: CREATED
- Receiver Type: (blank)
- Type linkage active: yes
- Enable event queue: deactivated

The type linkage indicator must not be active for all other receiver types of object type BUS2250 and event CREATED. This receiver type is defined via the receiver type function module USMD_WF_RECEIVER_TYPE. Make sure that receiver function module SWW_HI_CREATE_VIA_EVENT_IBF is entered.

Note
To enter the receiver type function module or if you need to change the settings, mark the according line in the table and choose Goto > Details.

2. Configure workflow tasks.
To configure workflow tasks, run the following activity in Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Workflow > Configure Workflow Tasks.

1. Go to application component CA-MDG-AF and choose Assign Agents.
2. All activities (denoted by TS*) that are not set as Background Task need to be set to General Task. To do so, select the activity, choose Attributes, and change to General Task.
3. Repeat the procedure for all non-background activities within the CA-MDG-APP-BP application component.

3. Check number of workflow steps
1. To define the workflow steps for the workflows assigned to your change request types (which shall be assigned to a processor), run the following activity in Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Workflow > Configure Workflow Steps.

   1. Go to application component CA-MDG-APP-BP and choose Configure Workflow Steps.
   2. The workflow steps must be assigned to the processor configured for your change request types.
   3. Repeat the procedure for all non-background activities within the CA-MDG-APP-BP application component.
Governance under ➤ General Settings ➤ Process Modeling ➤ Workflow ➤ Other MDG Workflows ➤ Define Change Request Step Numbers

2. If you use the change request types delivered with MDG-C, the following workflow steps are delivered:
   - WS54300003: Step 0, 1, 2, 3
   - WS46000023: Step 0, 1, 2, 3, 4, 5
   - WS46000027: Step 0, 1, 2, 3
   - WS60800095: Step 0, 1, 2, 3, 4, 5

4. Create organizational unit or change staff assignments for organizational unit
   Create an organizational unit with transaction PPOCW or change staff assignments for an organizational unit with transaction PPOME. Users who will process the workflow steps have to be assigned to this organizational unit.

5. Check or set up the business rule framework plus (BRFplus)
   1. To check the business rule framework plus (BRFplus) run the following activity in Customizing for Master Data Governance, Central Governance ➤ Master Data Governance for Customer ➤ Workflow ➤ Assign Processor to Workflow Step Number in BRFplus for Customer.

Note
If the system does not contain the Customizing application MDG_BS_ECC_CUSTOMER_WF_CUSTM export it from client 000 using transaction SCC1.

Note
To copy the content of the decision table GET_AGENT, do the following:
Prerequisite: In transaction BRF+ under Personalize the User Mode Expert is selected.
1. Run transaction BRF+ in the source client and search for Name MDG_BS_ECC_CUSTOMER_WF_CUSTM.
2. In the search result list expand MDG_BS_ECC_CUSTOMER_WF_CUSTM Expression and open GET_AGENT with a double click.
3. Choose Additional Actions ➤ Export to Excel to download the data.
4. In your target client repeat the steps 1 and 2 to open the decision table GET_AGENT.
5. Choose Additional Actions ➤ Import from Excel to upload the data.

2. To assign processors to workflow step numbers run the following activity in Customizing for Master Data Governance, Central Governance ➤ Master Data Governance for Customer ➤ Workflow ➤ Assign Processor to Workflow Step Number in BRFplus for Customer.
Assign processors, for example users or organizational units, for all change request types and their created workflow steps.

The following table shows an example of the change request types and their workflow steps.

<table>
<thead>
<tr>
<th>Change Request Type</th>
<th>Workflow Step Number</th>
<th>Object Type</th>
<th>Object ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUST1P2</td>
<td>01 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST1P2</td>
<td>04 Revision after Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
</tbody>
</table>
The change request type CUST1P2 supports distributed workflow for customers. It is delivered with the business function *Master Data Governance for Customer 7.0 Feature Pack* [external document].

For the change request types CUSTL1 (lean request UI for ERP customers) and CUSTL1P2 (distributed workflow for ERP customers) that are also delivered with the business function *Master Data Governance for Customer 7.0 Feature Pack* [external document] processors are assigned to workflow step numbers in the same way.

### 6. Assign processors to workflow steps (Simple Workflow)

1. To assign the object ID of the organizational units to the workflow step number, run the following activity in Customizing for [Master Data Governance, Central Governance ➤ General Settings ➤ Process Modeling ➤ Workflow ➤ Other MDG Workflows ➤ Assign Processor to Change Request Step Number (Simple Workflow)]

Assign processors, for example users or organizational units, for all change request types and their created workflow steps.

The following table shows an example of the change request types and their workflow steps.

<table>
<thead>
<tr>
<th>Change Request Type</th>
<th>Workflow Step Number</th>
<th>Object Type</th>
<th>Object ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUST2P1</td>
<td>01 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST2P1</td>
<td>02 Decision: Activation Despite Discrepancy</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST5P1</td>
<td>01 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST5P1</td>
<td>02 Decision: Activation Despite Discrepancy</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST5P1</td>
<td>03 Revision After Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST6P1</td>
<td>01 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST6P1</td>
<td>02 Decision: Activation Despite Discrepancy</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST6P1</td>
<td>03 Revision After Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUSTMRP1</td>
<td>01 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>Change Request Type</td>
<td>Workflow Step Number</td>
<td>Object Type</td>
<td>Object ID</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------</td>
<td>-------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>CUSTMRP1</td>
<td>02 Decision: Activation</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>Despite Discrepancy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUSTMRP1</td>
<td>03 Revision After Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUSTMRP2</td>
<td>01 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUSTMRP2</td>
<td>02 Decision: Activation</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>Despite Discrepancy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUSTMRP2</td>
<td>03 Revision After Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST1P3</td>
<td>01 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST1P3</td>
<td>02 Decision: Activation</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>Despite Discrepancy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUST1P3</td>
<td>03 Revision After Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST2P2</td>
<td>01 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUST2P2</td>
<td>02 Decision: Activation</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>Despite Discrepancy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUST2P2</td>
<td>03 Revision After Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUSTHI01</td>
<td>01 Maintenance of</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>Duplicate Entries</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUSTHI01</td>
<td>02 Approval (No Rejection)</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUSTHI01</td>
<td>03 Revision</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUSTHI01</td>
<td>04 Decision: Activation</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>Despite Discrepancy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUSTHI02</td>
<td>01 Approval (No Rejection)</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUSTHI02</td>
<td>02 Revision</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>CUSTHI02</td>
<td>03 Decision: Activation</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>Despite Discrepancy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUSTHI02</td>
<td>04 Revision of Address</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>After Validation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BPLP1</td>
<td>01 Processing</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>BPLP1</td>
<td>02 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>BPLP1</td>
<td>03 Revision after Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>BPLP1</td>
<td>04 Revision after Rejection</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>BPLP1</td>
<td>05 Decision: Activation</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td></td>
<td>Despite Discrepancy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BPMP1</td>
<td>01 Processing</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
<tr>
<td>BPMP1</td>
<td>02 Approval</td>
<td>O</td>
<td>OrgUnit-ID</td>
</tr>
</tbody>
</table>
### 3.8 Set Up the Rule-Based Workflow

**Note**
This document describes an example for a preconfigured Rule-Based Workflow for the Process Business Partner Cleansing Case.

MDG-C uses advanced workflow capabilities by combining the SAP Business Workflow with the SAP Business Rule Framework plus (BRFplus) tool. This section describes how to activate and load the necessary settings for BRFplus.

To use the predefined change request type BPCC2 run the following Customizing activity [Master Data Governance, Central Governance > Master Data Governance for Customer > Import Predefined Change Request Types](#) and activate the BC set CA-MDG-APP-CUS_VC_USMD110_C05.

Check that the steps for Rule-Based Workflow are imported by the BC-Set by running the following activity in Customizing: [General Settings > Process Modeling > Workflow > Rule-based Workflow > Define Change Request Steps for Rule-Based Workflow](#).

The following are the default steps required for the predefined change request type BPCC2.

<table>
<thead>
<tr>
<th>Type of Chg. Request</th>
<th>CR Step</th>
<th>Keys</th>
<th>Validation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPCC2</td>
<td>0</td>
<td></td>
<td></td>
<td>Processing</td>
</tr>
<tr>
<td>Type of Chg. Request</td>
<td>CR Step</td>
<td>Keys</td>
<td>Validation</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------</td>
<td>------</td>
<td>------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>BPCC2</td>
<td>90</td>
<td></td>
<td>X</td>
<td>Final Check</td>
</tr>
<tr>
<td>BPCC2</td>
<td>91</td>
<td></td>
<td></td>
<td>Activation</td>
</tr>
<tr>
<td>BPCC2</td>
<td>92</td>
<td></td>
<td></td>
<td>Revision</td>
</tr>
<tr>
<td>BPCC2</td>
<td>95</td>
<td></td>
<td></td>
<td>Revision Processing</td>
</tr>
<tr>
<td>BPCC2</td>
<td>96</td>
<td></td>
<td></td>
<td>Processing After Activation Error</td>
</tr>
<tr>
<td>BPCC2</td>
<td>99</td>
<td></td>
<td></td>
<td>Complete</td>
</tr>
</tbody>
</table>

To check that the predefined change request type exists see the Customizing activity [General Settings > Process Modeling > Change Requests > Create Change Request Type].

To check that the link between the predefined change request type and the generated BRF+ application ID exists, run transaction SE16, enter the table name USMD213C_SSW and look for the following entries:

<table>
<thead>
<tr>
<th>Table 10</th>
<th>USMD_CREQ_TYPE</th>
<th>USMD_BRFP_APP_ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPCC2</td>
<td>005056AC02D81ED2AC971CAB3C1DC848</td>
<td></td>
</tr>
</tbody>
</table>

Check that the BRF+ catalogs (inclusive application, functions, decision tables with content) for the predefined change request type are in the client.

Due to technical restrictions further settings currently cannot be included in the BC set processing therefore the following steps need to be executed manually:

1. Run transaction BRF+.
2. On the Catalog tab choose Select Catalog.
3. Enter the search criteria: Select the Object Type Catalog and as Name enter USMD_SSW_CATA_BPCC2.

**Note**

If the catalog does not exist in your client, export the catalog from the client 000 using transaction SCC1. The import will also include the values in the three pre-delivered decision tables. For testing the standard workflow configuration you have to start the workflow now.

Maintain your content in the decision tables, reflecting your governance process. You can use the pre-delivered SAP content as a starting point.

To maintain your content either run transaction USMD_SSW_RULE and select the correponding change request type or run the Customizing activity [General Settings > Process Modeling > Workflow > Rule-Based Workflow > Configure Rule-Based Workflow].

**Note**

If you extend the BRFplus decision tables make sure that the status of the final step is set to 05 Final Check Approved or 06 Final Check Rejected.

The following are the minimal settings for the relevant change request types.

- DT_SINGLE_VAL_BPCC2 (Single value decision table)
- DT_NON_USER_AGTR_GRP_BPCC2 (Non-user agend decision table)
DT_USER_AGT_GRP_BPCC2 (User agent decision table)

DT_NON_USER_AGT_GRP_BPCC2 and DT_SINGLE_VAL_BPCC2 contain the automated steps and follow-up logic used by the BRFplus. The follow-up logic can either result in an automated step or in a user task that will be made visible as a workflow.

The users involved in the Master Data Governance process need to be included in table DT_USER_AGT_GRP_BPCC2 so that they can receive and execute the workflow tasks. Maintain agents (for example the users or organizations in the Portal Content and Portal Role Assignment sections of this document) for all change request types and their associated workflow steps. Each condition alias that appears in the DT_SINGLE_VAL_BPCC2 table needs at least one processor, either automatic or user (group). Therefore, for all non-automated steps at least one entry needs to exist in DT_USER_AGT_GRP_BPCC2.

For further details on the workflow concepts, see Rule Based Workflow for Business Partner [external document]. You can adopt the settings described in that document for BP1P1 to BPCC2.

The following step types are pre-delivered by SAP:

<table>
<thead>
<tr>
<th>Step Type</th>
<th>Short Description</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Check Change Request</td>
<td>01 Agree; 02 Disagree</td>
</tr>
<tr>
<td>2</td>
<td>Approve Change Request</td>
<td>03 Approve; 04 Reject</td>
</tr>
<tr>
<td>3</td>
<td>Process Change Request</td>
<td>05 Finalize Processing; 06 Send for Revision</td>
</tr>
<tr>
<td>4</td>
<td>Revise Change Request</td>
<td>07 Resubmit; 08 Withdraw</td>
</tr>
<tr>
<td>5</td>
<td>Activate Change Request</td>
<td>09 Activate; 10 Send for Revision</td>
</tr>
<tr>
<td>6</td>
<td>Approve Change Request (Without Rejection)</td>
<td>03 Approve; 06 Send for Revision</td>
</tr>
<tr>
<td>7</td>
<td>Revise Change Request (Without Rejection)</td>
<td>05 Finalize Processing</td>
</tr>
<tr>
<td>8</td>
<td>Activate Change Request</td>
<td>09 Activate; 04 Reject</td>
</tr>
</tbody>
</table>

The following agent types exist:

<table>
<thead>
<tr>
<th>User Agent Type</th>
<th>Short Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>US</td>
<td>User</td>
</tr>
<tr>
<td>0</td>
<td>Organizational Unit</td>
</tr>
<tr>
<td>C</td>
<td>Job</td>
</tr>
<tr>
<td>S</td>
<td>Position</td>
</tr>
<tr>
<td>AG</td>
<td>Role</td>
</tr>
<tr>
<td>SU</td>
<td>Special User (Initiator/Last-Step User)</td>
</tr>
</tbody>
</table>

The following is the standard content pre-delivered by SAP:

DT_SINGLE_VAL_BPCC2
Table 13

<table>
<thead>
<tr>
<th>Previous Change Request Step (PREVIOUS_STEP)</th>
<th>Previous Action (PREVIOUS_ACTION)</th>
<th>Condition Alias (COND_ALIAS)</th>
<th>New Change Request Step (NEW_STEP)</th>
<th>New Change Request Status (NEW_CR_STATUS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 (Processing)</td>
<td>No value</td>
<td>1</td>
<td>90 (Final Check)</td>
<td>01 (To Be Considered and Approved)</td>
</tr>
<tr>
<td>90 (Final Check)</td>
<td>03 (Approve)</td>
<td>2</td>
<td>91 (Activation)</td>
<td></td>
</tr>
<tr>
<td>90 (Final Check)</td>
<td>04 (Reject)</td>
<td>3</td>
<td>95 (Revision Processing)</td>
<td>10 (To Revise: Perform Changes)</td>
</tr>
<tr>
<td>91 (Activation)</td>
<td>31 (Activation Successful)</td>
<td>4</td>
<td>99 (Complete)</td>
<td>05 (Final Check Approved)</td>
</tr>
<tr>
<td>91 (Activation)</td>
<td>&lt;&gt;31 (Activation Successful)</td>
<td>5</td>
<td>96 (Processing After Activation Error)</td>
<td>11 (Process Errors After Activation)</td>
</tr>
<tr>
<td>92 (Revision)</td>
<td>No value</td>
<td>6</td>
<td>99 (Complete)</td>
<td>06 (Final Check Rejected)</td>
</tr>
<tr>
<td>95 (Revision Processing)</td>
<td>07 (Resubmit)</td>
<td>1</td>
<td>90 (Final Check)</td>
<td>01 (To Be Considered and Approved)</td>
</tr>
<tr>
<td>95 (Revision Processing)</td>
<td>08 (Withdraw)</td>
<td>8</td>
<td>92 (Revision)</td>
<td></td>
</tr>
<tr>
<td>96 (Processing After Activation Error)</td>
<td>09 (Activate)</td>
<td>9</td>
<td>91 (Activation)</td>
<td></td>
</tr>
<tr>
<td>96 (Processing After Activation Error)</td>
<td>10 (Send for Revision)</td>
<td>3</td>
<td>95 (Revision Processing)</td>
<td>10 (To Revise: Perform Changes)</td>
</tr>
</tbody>
</table>

DT_USER_AGT_GRP_BPCC2

Table 14

<table>
<thead>
<tr>
<th>Condition Alias (COND_ALIAS)</th>
<th>User Agt Grp No. (AGENT_GROUP)</th>
<th>Step Type (STEP_TYPE)</th>
<th>User Agent Type (USER_TYPE)</th>
<th>User Agent Value (USER_VALUE)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>001</td>
<td>2</td>
<td>SU (Special User (Initiator/Last-Step User))</td>
<td>INIT</td>
</tr>
<tr>
<td>3</td>
<td>001</td>
<td>4</td>
<td>SU (Special User (Initiator/Last-Step User))</td>
<td>INIT</td>
</tr>
<tr>
<td>5</td>
<td>001</td>
<td>5</td>
<td>SU (Special User (Initiator/Last-Step User))</td>
<td>INIT</td>
</tr>
</tbody>
</table>

DT_NON_USER_AGT_GRP_BPCC2
### Table 15

<table>
<thead>
<tr>
<th>Condition Alias</th>
<th>Agent Group</th>
<th>Process Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>COND_ALIAS</td>
<td>AGENT_GROUP</td>
<td>PROCESS_PATTERN</td>
</tr>
<tr>
<td>2</td>
<td>001</td>
<td>05 (Activation (Do Not Bypass Snapshot))</td>
</tr>
<tr>
<td>4:6</td>
<td>001</td>
<td>99 (Complete (Sub-)Workflow)</td>
</tr>
<tr>
<td>8</td>
<td>001</td>
<td>08 (Roll back change request)</td>
</tr>
<tr>
<td>9</td>
<td>001</td>
<td>06 (Activation (Bypass Snapshot))</td>
</tr>
</tbody>
</table>

**Note**
- It is necessary to activate your new entries for each of the three decision tables.
- In the preconfigured delivery for all step types the value for user agent type is SU (Special User) and the user agent value is INIT (initiator). Therefore testing is immediately possible.
- The quickest way to allow additional users to participate for example in the approve step for testing purposes is to copy the default entry for step type 2 into the DT_USER_AGT_GRP_BPCC2 table and change the entries as follows:
  - User Type: AG (Role)
  - User Value: SAP_MDGC_MENU_04 (The MDG-C role of PFCG/SU01)
- If the system cannot find a processor for your change requests check the background steps of the change requests workflow log to see if a work item with the description Set Status x for Change Request y is in process or in error. If this is the case regenerate the authorization profile of SAP_ALL or include USMD* authorization objects into the authorization for the user WF-Batch.
- Choosing the process pattern 06 Activation (Bypass Snapshot) means that the business partner will be activated, even if the business partner record was changed in the backend system since the change request was created. Any backend changes are lost upon activation. You can adjust this behavior with SAP Note 1797009. Implementing this note means that the system only overwrites changes to entities present in the change request rather than all entities.

### 3.9 Show Master Data Governance for Customer 6.1 Specific UIBBs

To show the user interface building blocks (UIBB) for the entities delivered with the business function Master Data Governance for Customer 6.1 proceed as follows:

**Note**
- Depending on whether you use the customer UI or the UI configuration BS_OVP_BP_ALL carry out the corresponding procedure.

**Prerequisites:**
- In transaction SU01 on the Parameters tab the Set/Get parameter FPM_CONFIG_EXPERT is set to the value A.
• The authorization objects S_DEVELOP and S_WDR_P13N are assigned to your user.

Show specific UIBBs for the Customer UI

1. Run transaction NWBC, enter the role SAP_MDGC_MENU_04 and choose Start NetWeaver Business Client for HTML.
2. Choose Search Customer.
3. Choose (Customize Page).

   Note
   If an error message is displayed concerning missing object component Customizing proceed as follows:
   1. Choose New.
   2. In the Create Customizing dialog box enter a description and choose OK.
   3. In the Select Transport Request dialog box choose OK.

4. Mark Main Page BS_BP_OVP.
5. On the Overview Page Schema tab, choose (Expand Node) in front of Section: BP_SECTION to open the UIBB structure.
6. Mark UIBB: Relationship BS_BP_RELATIONS.
7. In the Hidden Element field in the Attributes of UIBB section select Visible and choose Save.
8. Repeat the procedure as of step 4 using the following values:
   ○ Edit Page BS_CU_MLT_ASSIGNMENT
      ○ UIBB: ERP Customer: Company Codes BS_CU_COMPANY_CODES
      ○ UIBB: ERP Customer: Sales Area BS_CU_SALES AREAS
      ○ UIBB: ERP Customer: Tax Indicators BS_CU_TAX_INDICATORS
9. As a final step on the Search Customer screen choose (Personalize) and on the Personalize dialog box choose Reset to Default.

   Note
   To show the corresponding UIBBs within Block and Mark for Deletion change requests, mark an existing customer in the search result list, choose Block or Mark for Deletion and proceed as described above as of step number 3.

Show specific UIBBs for the ERP Customer UI

1. Run transaction SE80, select the package MDG_BS_BP_BOLUI.
2. Depending on your SAP NetWeaver release choose:
   ○ SAP NetWeaver 7.31: Web Dynpro Web Dynpro Applicat. BS_OVP_BP Applic. Configurations BS_OVP_BP_ALL
   ○ SAP NetWeaver 7.40: Web Dynpro FPM Applications BS_OVP_BP FPM Application Configurations BS_OVP_BP_ALL
3. Double-click BS_OVP_BP_ALL and choose Test/Execute.
Note

If an error message is displayed concerning missing object component Customizing proceed as follows:

1. Choose New.
2. In the Create Customizing dialog box enter a description and choose OK.
3. In the Select Transport Request dialog box choose OK.

5. Mark Main Page BS_BP_OVP.
6. On the Overview Page Schema tab, choose ▶ (Expand Node) in front of Section: BP_BP_OVP to open the UIBB structure.
7. Mark UIBB: Relationship BS_BP_RELATIONS.
8. In the Hidden Element field in the Attributes of UIBB section select Visible and choose Save.
9. Repeat the procedure as of step 4 using the following values:
   ○ Edit Page BS_CU_MLT_ASSIGNMENT
   ○ UIBB: ERP Customer: Company Codes BS_CU_COMPANY_CODES
   ○ UIBB: ERP Customer: Sales Area BS_CU_SALES_areas
   ○ UIBB: ERP Customer: Tax Indicators BS_CU_TAX_INDICATORS
10. As a final step on the Search Business Partner screen choose (Personalize) and on the Personalize dialog box choose Reset to Default.

3.10 Show Master Data Governance for Customer 9.0 Specific UIBBs

To show the user interface building blocks (UIBB) for the entities delivered with the business function Master Data Governance for Customer 9.0 proceed as follows:

Note

Depending on whether you use the customer UI or the UI configuration BS_OVP_BP_ALL, carry out the corresponding procedure.

Prerequisites:

- In transaction SU01 on the Parameters tab, the Set/Get parameter FPM_CONFIG_EXPERT is set to the value A.
- The authorization objects S_DEVELOP and S_WDR_P13N are assigned to your user.

Show specific UIBBs for the Customer UI

1. Run transaction NWBC, enter the role SAP_MDGC_MENU_04 and choose Start NetWeaver Business Client for HTML.
2. Choose Search Customer.
3. Choose (Customize Page).
Note
If an error message is displayed concerning missing object component Customizing proceed as follows:
1. Choose New.
2. In the Create Customizing dialog box, enter a description and choose OK.
3. In the Select Transport Request dialog box, choose OK.

4. Mark Main Page BS_BP_OVP.
5. On the Overview Page Schema tab, choose (Expand Node) in front of Section: BP_SECTION to open the UIBB structure.
6. Mark UIBB: Hierarchy Assignments BS_BP_HRY_ASSIGNMENTS.
7. In the Hidden Element field in the Attributes of UIBB section, select Visible and choose Save.
8. Repeat the procedure as of step 4 using the following values:
   ○ Edit Page BS_CU_MLT_ASSIGNMENT
     ○ UIBB: ERP Customer: Texts BS_CU_LONGTEXT_CU
   ○ Edit Page BS_CU_COMPANY_CODE
     ○ UIBB: ERP Customer: Company Code: Texts BS_CU_COMPANY_CODE_LONGTEXT_CU
   ○ Edit Page BS_CU_SALES_AREA
     ○ UIBB: ERP Customer: Sales Area: Texts BS_CU_SALES_AREA_LONGTEXT_CU
   ○ Edit Page BS_BP_RELATION_PARTNER
     ○ UIBB: Partner Details BS_BP-details_REL

   Note
As an additional step to show this UIBB, you have to hide the UIBB that is shown by default as follows:
Mark UIBB: Partner Details BS_BP_DETAILS.
In the Hidden Element field in the Attributes of UIBB section, select Hidden and Excluded from Event Loop and choose Save.

   ○ Edit Page BS_BP_WORKPLACE
     ○ UIBB: International Versions BS_BP_WORKPLACE_VERSIONS
9. As a final step on the Search Customer screen choose (Personalize) and on the Personalize dialog box choose Reset to Default.

   Note
To show the corresponding UIBBs within Block and Mark for Deletion change requests, mark an existing customer in the search result list, choose Block or Mark for Deletion and proceed as described above as of step number 3.

Show specific UIBBs for the ERP Customer UI
1. Run transaction NWBC, enter the role SAP_MDGC_CL_MENU_04 and choose Start NetWeaver Business Client for HTML.
2. Choose Search Customer.
3. Choose 📚 (Customize Page).

**Note**

If an error message is displayed concerning missing object component Customizing proceed as follows:

1. Choose **New**.
2. In the Create Customizing dialog box, enter a description and choose **OK**.
3. In the Select Transport Request dialog box, choose **OK**.

4. Mark **Main Page** BS_BP_OVP.

5. On the **Overview Page Schema** tab, choose ▶ (Expand Node) in front of **Section: BP_SECTION** to open the UIBB structure.

6. Mark **UIBB: Texts** BS_CU_LONGTEXT_CU.

7. In the **Hidden Element** field in the **Attributes of UIBB** section, select **Visible** and choose **Save**.

8. Repeat the procedure as of step 4 using the following values:
   - **Edit Page** BS_CU_CL_CONTACT
   - **UIBB: International Versions** BS_BP_WORKPLACE_VERSIONS
   - **Edit Page** BS_CU_COMPANY_CODE
   - **UIBB: ERP Customer Company Code: Texts** BS_CU_COMPANY_CODE_LONGTEXT_CU
   - **Edit Page** BS_CU_SALES_AREA
   - **UIBB: ERP Customer Sales Area: Texts** BS_CU_SALES_AREA_LONGTEXT_CU

9. As a final step on the **Search Customer** screen choose 📚(Personalize) and on the **Personalize** dialog box choose **Reset to Default**.

**Note**

To show the corresponding UIBBs within **Block** and **Mark for Deletion** change requests, mark an existing customer in the search result list, choose **Block** or **Mark for Deletion** and proceed as described above as of step number 3.

### 3.11 Set Up Business Partner

You set up the business partner Customizing and check number ranges and groupings that are mandatory for MDG-C.

In addition if you need to set up or define settings for SAP Business Partner, run the following activity in Customizing for **Master Data Governance, Central Governance** under [» Master Data Governance for Business Partner » Set up Business Partner »].

**Note**

Make sure your Customizing for the object **Business Partner** is harmonized throughout your hub and your client systems.
3.12 Set Up Customer

You set up or check the Customizing for the customer master in the MDG hub system and for the relevant client systems.

If you need to set up the customer master, perform the following activity in Customizing for Master Data Governance, Central Governance under Master Data Governance for Customer Integration with Customer Master in ERP Set up Customer Master for Master Data Governance for Customer.

**Note**

Make sure your Customizing for the object Customer is harmonized throughout your hub and your client systems.

3.13 Set Up Customer-Vendor Integration

You define the settings for Customer-Vendor Integration (CVI) in Customizing for Master Data Governance, Central Governance under Master Data Governance for Customer Integration with Customer Master in ERP Set up Customer Vendor Integration for MDG for Customer.

3.14 Replicate Data

Replication of master data from the MDG hub to the connected systems and clients can be done using the following methods:

- Data Replication Using Enterprise Service Oriented Architecture [page 131]
- Data Replication of Customer Master Data Using ALE [page 132]
- Data Replication of Business Partner Master Data Using ALE [page 136]

**Note**

For information on how to replicate data to a CRM system, see Configuration for Data Replication to CRM Systems [page 140].

3.15 Define Value Mapping

If required, you can define value mapping for elements such as, BP relationship role code and others in Customizing for Master Data Governance, Central Governance under General Settings Value Mapping.

The fields of the business partner record for which a value mapping can be defined, are stored in the table MDGV_ELEMENT.
More Information

For further information on value mapping, see Value Mapping [page 145].

3.16 Define Key Mapping

If you are working with multiple connected systems and did not consolidate the business partner keys during the initial load phase, key mapping may be required. You can define the system-specific mappings for the key value of the business partner in Customizing for Master Data Governance, Central Governance under General Settings → Key Mapping.

3.17 Search and Duplicate Check

To configure the search and the duplicate check run the Customizing-activities under Master Data Governance, Central Governance → General Settings → Data Quality and Search → Search and Duplicate Check.

3.18 Data Quality Services

You have the option to integrate SAP Master Data Governance with data quality management solutions, such as SAP Analytics Data Quality Management, including address validation, duplicate check, and data enrichment. While creating new master data records, for example business partners, you can use the Web Dynpro application MDG_ADDR_CHECK_TEST (application configuration MDG_ADDR_CHECK_TEST_AC) to check that the entered address is valid and complete. This function supports you in detecting inconsistencies. It also helps you to improve the quality and completeness of your master data records with standardized address information on country and region codes, postal codes, as well as street names and house numbers. Alternatively, you can also integrate third-party solutions that offer, for example, updating of databases, change of address service, address enrichment, or data cleansing.

For more information, see Validation and Enrichment [external document].

For more information about data quality, see SAP Help Portal at help.sap.com Analytics All Products.

3.19 Set Up Embedded Search

Prerequisites

- TREX is setup and configured for the Embedded Search in the system before enabling the master data object for Embedded Search.
  For further information how to setup TREX, see Creating a Connection Between Embedded Search and TREX/BWA [external document].
- You are authorized to access the Connector Administration Cockpit (Transaction ESH_COCKPIT).
**Procedure**

Once TREX is configured, follow the steps mentioned below to enable the master data object for Embedded Search:

- **Note**
  
  The search object connector template for business partner (MDG_BUSINESS_PARTNER) contains information about entities and attributes and about the relationship among entities as defined in the data model.

Configure the search connector using transaction ESH_COCKPIT. For further information, see Using the Connector Administration Cockpit [external document].

Create a search connector for the corresponding embedded search template (MDG_BUSINESS_PARTNER).

- After creating the search connector, an initial extraction of data for the connector is possible, choosing Actions Schedule Indexing. This will select all data in database tables corresponding to a master data object.
- Due to data changes after the initial extraction the embedded search index needs to be updated. To set this option choose Actions Schedule Indexing and select Real-time Indexing.

### 3.20 Configuring SAP HANA-Based Search for MDG

SAP HANA-based search for SAP Master Data Governance enables you to perform searches and duplicate checks on master data residing in the SAP HANA database. An SAP HANA search provider is delivered to enable these features.

The following data models are supported out-of-the-box for MDG on HANA:

- Flex data models
- The business partner reuse model (BP)
- The material reuse model (MM)

The access class implementation is not provided for other reuse models. You must implement the access class for SAP HANA search to use it with the other reuse models.

SAP HANA-based search for SAP Master Data Governance can be used for the following MDG applications:

- Master Data Governance for Custom Objects
- Master Data Governance for Financials
- Master Data Governance for Supplier
- Master Data Governance for Customer
- Master Data Governance for Material

### Prerequisites

You have activated the business functions Master Data Governance, Generic Functions 7.0 or higher (MDG_FOUNDATION_4 or higher) and Master Data Governance, Generic Functions 7.0 Feature Pack or higher (MDG_FOUNDATION_5 or higher).

You have installed the SAP HANA database, support package 06 or higher. We recommend that you install the highest available version of the SAP HANA database.
You must have the following permissions to work with search views in SAP HANA:

- Permission to create a package and to write objects into packages
- Permission to create, change and drop attribute views
- Permission to create, change and drop SQL views
- Permission to create, execute and drop rule sets

For more details please refer to the SAP HANA security guide.

Process

To configure SAP HANA-based search for MDG, carry out the steps described below.

1. **Create Database Connection**

   Run transaction DBCO and create a database connection to the SAP HANA database.

   

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Connection Name</td>
<td>Unique name for the SAP HANA database connection used for search and duplicate check</td>
</tr>
<tr>
<td>Database System</td>
<td>SAP HANA database</td>
</tr>
<tr>
<td>Permanent</td>
<td>Yes</td>
</tr>
<tr>
<td>User Name</td>
<td>Schema name created in step above</td>
</tr>
<tr>
<td>Connection Information</td>
<td>Server: instance number</td>
</tr>
<tr>
<td>Connection Limit</td>
<td>0</td>
</tr>
<tr>
<td>Optimum Number of Connections</td>
<td>0</td>
</tr>
</tbody>
</table>

   **Note**

   **Deployment Options for MDG**

   MDG can be deployed on an SAP HANA database or on any database.
   - If you deploy MDG on SAP HANA, then SAP HANA acts as the primary database and the creation of the database connection is optional. If the database connection is not maintained then a default connection is derived automatically.
   - If you deploy MDG on any other database, then you must maintain the database connection to the schema in the SAP HANA database.

2. **Maintain the MDG SAP HANA Database Profile Settings**

   You must define the MDG landscape settings, such as the connection to the SAP HANA database that is used for the search and duplicate check processes. You can make these setting in Customizing under Master Data Governance, Central Governance > General Settings > Technical Settings for Master Data > Define MDG Landscape Profile.

   The use of an SAP Landscape Transformation (SLT) server is optional for MDG data replication. If you use SLT for replicating the MDG table data to the SAP HANA database system, then you must also define a connection to an SLT server as explained below.
**Note**

**Deployment Options for MDG**

- If you deploy MDG on *SAP HANA*, then *SAP HANA* acts as the primary database and no replication is required.
- If you deploy MDG on another database, the MDG data must be replicated to *SAP HANA* search schema either by *SAP Landscape Transformation* (SLT) or by other means. To generate a search view in the target system where search is performed, the MDG table metadata and data must be replicated to the *SAP HANA* database. To enable this initial replication of the data you must carry out the steps described below.

1. Run transaction MDGIMG.
2. Navigate to **Master Data Governance, Central Governance ➤ General Settings ➤ Technical Settings for Master Data ➤ Define MDG Landscape Profile**.
3. Enter data in the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Database Connection Name for MDG</strong></td>
<td>The <em>SAP HANA</em> database used for the search and duplicate check processes created in the previous step. This field is optional if MDG is deployed on a <em>SAP HANA</em> database.</td>
</tr>
<tr>
<td><strong>RFC Connecting MDG to SLT System</strong></td>
<td>Optional, only enter data if you use SLT for data replication</td>
</tr>
<tr>
<td><strong>SLT Configuration Name</strong></td>
<td>Optional, only enter data if you use SLT for data replication</td>
</tr>
</tbody>
</table>

In the *SAP HANA* system, where the search on MDG data is performed, you must generate the search view. If you deploy MDG on a traditional database, and use SLT for replication then, when generating the view, before it is created, the system replicates the required table metadata to the *SAP HANA* database using the SLT settings.

If *SAP HANA* is the primary database, it is not mandatory to maintain the database connection name in MDG Landscape Profile customizing. If the name is not maintained the system uses the default database connection. You still have the option of maintaining a different connection name in the MDG Landscape Profile if you do not wish to use the default database connection.

4. In the SLT system the SLT user requires the authorization object *S_DMIS*, with the following field values defined for their role:

<table>
<thead>
<tr>
<th>Authorization Object</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Activity (ACTVT)</strong></td>
<td>02 (Change)</td>
</tr>
<tr>
<td><strong>MBT PCL: Scenario (MBT_PR_ARE)</strong></td>
<td>SLOP (SAP Landscape Transformation)</td>
</tr>
<tr>
<td><strong>MBT PCL: Processing Role Level (MBT_PR_LEV)</strong></td>
<td>PACKAGE (Transfer package level)</td>
</tr>
</tbody>
</table>

5. For Material Search, in transaction SA38 execute the report **MDG_HDB_MAT_MIGRATE_LONGTEXT** as a background job. Select the **Overwrite target table records** checkbox, to perform the initial load of material...
long texts to the database table MDGHDB_LONGTEXT. This loads the following long text types: Basic Data Text, Sales Text, Purchase Order Text, Inspection Text, Internal Note, and Material Note MRP.

3. Define Authorization Relevance for Each Entity Type
   The authorizations maintained in customizing are considered during search. You can maintain the authorization in Customizing under [Master Data Governance, Central Governance ➤ General Settings ➤ Data Modeling ➤ Define Authorization Relevance per Entity Type] in the Business Modeler tool.

4. Create Search View
   Create a search view in the development system and transport it to the test and production systems. The search view must be generated or regenerated in the target (test and production) systems.
   If you are using the business partner, customer, or supplier domains and have activated the business functions MDG_ERP_CUSTOMER_3 or higher, MDG_ERP_SUPPLIER_4 or higher, or MDG_BUPA_1 or higher, or if you are using material domain and have activated the business function MDG_MATERIAL_5 or higher, then you must assign the template views from these business functions to a customer package in the Create Search View configuration activity before you can generate and use them.
   You must also have authorization to create a workbench request.
   To create a search view, run transaction MDG_HDB_GEN_UI or navigate to [Master Data Governance, Central Governance ➤ General Settings ➤ Data Quality and Search ➤ Search and Duplicate Check ➤ Create Search View].
   The package where you generate the search view must be in the customer namespace. Enter the name of the package during search view creation.
   When you create the search view and the system generates the SAP HANA view, the following search configuration data is automatically updated:
   ○ [Master Data Governance, Central Governance ➤ General Settings ➤ Data Quality and Search ➤ Search and Duplicate Check ➤ Define Search Applications ➤ Allocation of Search Help to Search Applications]
   ○ [Master Data Governance, Central Governance ➤ General Settings ➤ Data Quality and Search ➤ Search and Duplicate Check ➤ Define Search Applications ➤ Allocation of Entities to Search Help]
   In the SAP HANA studio, the search view is regenerated, the new search view will overwrite the manually updated search view.
   You can create a search rule set during the search view generation if you want the search to be performed based on search rule sets. If you choose the create ruleset option for a reuse model, a union SQL view is created on the attribute view in SAP HANA. This search rule set can also be manually updated according to the business requirements of the users after it is generated. If the search view is edited at a later date and is regenerated, the search view will not be regenerated/overwritten; it has to be manually adjusted.
   You must manually check out the generated search rule set to the Project Explorer view of the SAP HANA Studio Administration Console before it can be edited to change any parameter, such as the fuzzy value or weight of an attribute, and activate it to enable search based on this modified search rule set.
   You can also copy an existing search view and edit it before generating the search view.
   If there is a mismatch between the generated search view and the underlying objects, the system recognizes this and updates the status of the generated search view to Outdated. You can edit this outdated search view and regenerate the view.
   To delete a search view, you must first remove the customizing settings for the search view, and then delete the search view. The status of the view is then set to Marked for Deletion. In transaction SE38 execute the report program MDG_HDB_DELETE_SEARCH_VIEWS to delete the specific view or all views that are marked for deletion, and drop the corresponding objects in SAP HANA.
   You must set filters in the SAP HANA staging views to exclude records that have the obsolete indicator set.
   Identify all the Obsolete Indicator flags. The fields corresponding to the obsolete indicator flags in each table...
of a staging view have the technical naming convention USMD*_OBS_* or USMD*_O_* . Select the obsolete indicator in the Details section of the staging view, right click and select Apply Filter. In the Operator field select Not Equal and in the Value field enter X.

Table 19

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Operator</th>
<th>Filter Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>USMD<em><em>OBS</em></em></td>
<td>Not Equal</td>
<td>X</td>
</tr>
<tr>
<td>USMD<em><em>O</em></em></td>
<td>Not Equal</td>
<td>X</td>
</tr>
</tbody>
</table>

For material search you must set filters in the SAP HANA views for the material-related long texts stored in the database table MDGHDB_LONGTEXT. This means that only the appropriate long texts are taken from MDGHDB_LONGTEXT. To do this, in the SAP HANA studio, open the Content folder and navigate to the package where the search views are created. For reuse entity types, creating a search view generates two views in the SAP HANA system (one each for the active and staging areas), or three if you are using classification data. The views generated for the active area have names similar to searchviewname_Reuse and searchviewname_RINOB.

Open the reuse SAP HANA views below. Go to Detail window, and select the long text table with the alias you want to update and right-click on the attribute. From the menu choose Apply Filter. From the drop-down menu choose the operator Equal and maintain the values as specified in the tables below.

Basic Text

Table 20

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Filter Value</th>
<th>Table Name (Alias)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSCDATTXT_TDID</td>
<td>GRUN</td>
<td>BSCDATTXT_MDGHDB_LONGTEXT</td>
</tr>
<tr>
<td>BSCDATTXT_TDOBJECT</td>
<td>MATERIAL</td>
<td>BSCDATTXT_MDGHDB_LONGTEXT</td>
</tr>
</tbody>
</table>

Sales Text

Table 21

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Filter Value</th>
<th>Table Name (Alias)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SALESTXT_TDID</td>
<td>0001</td>
<td>SALESTXT_MDGHDB_LONGTEXT</td>
</tr>
<tr>
<td>SALESTXT_TDOBJECT</td>
<td>MVKE</td>
<td>SALESTXT_MDGHDB_LONGTEXT</td>
</tr>
</tbody>
</table>

Quality Inspection Text

Table 22

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Filter Value</th>
<th>Table Name (Alias)</th>
</tr>
</thead>
<tbody>
<tr>
<td>QINSPTXT_TDID</td>
<td>PRUE</td>
<td>QINSPTXT_MDGHDB_LONGTEXT</td>
</tr>
<tr>
<td>QINSPTXT_TDOBJECT</td>
<td>MATERIAL</td>
<td>QINSPTXT_MDGHDB_LONGTEXT</td>
</tr>
</tbody>
</table>

Purchase Text

Table 23

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Filter Value</th>
<th>Table Name (Alias)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PURCHTXT_TDID</td>
<td>BEST</td>
<td>PURCHTXT_MDGHDB_LONGTEXT</td>
</tr>
<tr>
<td>PURCHTXT_TDOBJECT</td>
<td>MATERIAL</td>
<td>PURCHTXT_MDGHDB_LONGTEXT</td>
</tr>
</tbody>
</table>
5. **Verify Customizing Settings for Search View**

After you have created and saved the search view, you must verify that the customizing settings are automatically updated for the newly created search view. To do this, perform the following:

1. Run transaction `MDGIMG`.
2. Navigate to Master Data Governance, Central Governance > General Settings > Data Quality and Search > Search and Duplicate Check > Define Search Applications.
3. Select the row with the Search Mode HA (HANA). Note that the Fuzzy checkbox has no impact on SAP HANA search; SAP HANA search is fuzzy by default and this cannot be changed here.
4. Double-click on Allocation of Search Help to Search Applications.
5. Verify that there is an entry for the newly created search view in the Included Search Help field with the technical name provided during search view creation.
6. Select the row of the newly created search view.
7. Double-click on Allocation of Entities to Search Help and verify that the main entity type that you selected during search view creation is updated.

6. **Create Match Profile for Duplicate Checks based on SAP HANA Search**

If you have created a search rule set in the Create Search View step, you can use it to configure the match profile for duplicate checks.

1. Run transaction `MDGIMG`.
2. Navigate to Master Data Governance, Central Governance > General Settings > Data Quality and Search > Search and Duplicate Check > Define Search Applications.
3. Select the row with the Search Mode HA (HANA).
4. Double-click on Match Profile.
5. For the specific data model and the Match Profile ID for Duplicate Check enter the name of the search rule set if you generated one in step 4 above, otherwise, leave the field empty.

When you enter the search rule set name, the information from the search rule set is used instead of the attribute view while performing search during duplicate checks.

7. **Configure Duplicate Check Based on SAP HANA search**

After you have maintained a match profile ID, you can configure the search view for duplicate checks.

1. Execute transaction `MDGIMG`.
2. Navigate to Master Data Governance, Central Governance > General Settings > Data Quality and Search > Search and Duplicate Check > Configure Duplicate Check for Entity Types.
3. Select the **Data Model** and **Entity Type** for which you want to configure the duplicate check. Select the **Search Mode** as HA. Enter the threshold values for the duplicate check. Enter the name of the **Match Profile ID** and the search view to be used for the duplicate check. Select the **Match Profile Based UI** if required.

8. **Test Search and Duplicate Check**
   1. Test the SAP HANA Search
      1. Launch the SAP NetWeaver Business Client.
      2. Select the work center for your data model.
      3. Launch the search UI.
      4. In the **Search Method** field enter the SAP HANA Search Configuration that you have created.
      5. Choose **Search** and the search results should be returned.
      6. In addition, perform a freestyle search and an attribute search and check the results.
   2. Test the Duplicate Check
      1. Create a duplicate of an existing object.
      2. When you have entered data for your object choose **Check**. This triggers the duplicate check and the system should indicate that your new object is a potential duplicate.

**Result**

You have now configured your system to use SAP HANA for MDG search. For drill down search configuration, see **Configuring Drill-Down Search (Optional)** [external document].

### 3.21 Choose the UI environment you want to use to run SAP Master Data Governance

You can run SAP Master Data Governance in the environments of the SAP NetWeaver Business Client or in the environment of the SAP NetWeaver Portal.

**SAP NetWeaver Business Client**

If you are running SAP Master Data Governance on the SAP NetWeaver Business Client (and not on the SAP NetWeaver Portal), you need to configure the roles for the Business Client in the SAP ERP system. To do so, perform the following steps in the SAP ERP system:

1. On the SAP Easy Access screen, choose **Tools ➤ Administration ➤ User Maintenance ➤ Role Administration ➤ Roles** and choose menu role **SAP_MDGC_MENU_04**. Alternatively, use transaction **PFCG** (Role Maintenance).

   Assign the menu role **SAP_MDGC_MENU_04** to your users.

   **SAP_MDGC_MENU_04, Master Data Governance for Customer: Menu** [external document]

1. **Note**

   If you use the ERP Customer UI assign the role **SAP_MDGC_CL_MENU_04, Master Data Governance for Customer (ERP Customer UI): Menu** [external document].
2. On the SAP Easy Access screen, choose Tools > Administration > User Maintenance > Users. Alternatively use transaction SU01 (User Maintenance) and assign the authorization role or the name of the copied role for the application to the master data governance user.

The following authorization roles are delivered:

- SAP_MDGC_DISP_05, Master Data Governance for Customer: Display [external document]
- SAP_MDGC_REQ_05, Master Data Governance for Customer: Requester [external document]
- SAP_MDGC_SPEC_05, Master Data Governance for Customer: Specialist [external document]
- SAP_MDGC_STEW_04, Master Data Governance for Customer: Data Steward [external document]

Run transaction PFCG to verify the setting of the authorization objects within the roles and adapt the authorizations to your organizational structures.

SAP NetWeaver Portal

The SAP NetWeaver Portal content for MDG-C is derived directly from the system PFCG roles. To create SAP NetWeaver Portal roles for your users, you must log on to your portal and upload the content information from your back-end system PFCG roles.

To upload your portal content to the portal, do the following:

1. Set up your SAP NetWeaver Portal for MDG.
2. In the Content Administration work center choose Portal Content Management > Portal Content and select a portal content folder to upload the portal content.
3. Right-click on the folder and choose New > Role > Role from Back End.
4. Select the system and client (or the connected system alias) you want to upload the role information from. This should be your hub system.
5. From the list displayed select the PFCG role SAP_MDGC_MENU_04 and begin the upload.

Once the MDG portal roles have been uploaded, you must assign them as follows:

1. Log on to the portal.
2. Choose Delegated User Administration.
3. Enter your User ID and choose Go.
4. Mark the line of your user and choose Modify.
5. Select the Assigned Roles tab.
6. Enter MDG as the search criteria.
7. Select the portal role you have uploaded before.
8. Choose Add and save.

After assigning the user role you need to log off and log on again to the portal. For more information on uploading role information, see SAP Note 1685257.

Note
To enable the navigation from FPM (Floorplan Manager) search results to the corresponding object maintenance, see SAP Note 1879171.
3.22 Data Transfer of Customer Master Data

Data transfer allows you to move master data between systems. These systems can be client or your main Master Data Governance systems.

If MDG-C is used on a productive ERP system in the same client no further initial import is required.

If MDG-C is used on a standalone ERP system customer data that is not available on the ERP hub needs to be imported from one of the following sources:

- Another ERP system
- A CRM system
- An SRM system
- A non-SAP system

Data Export from Source System

To be able to export customer master data, you can use the Export Master Data [external document] service (SAP ERP 6.0, EHP6). For exporting customer master data from systems with ERP releases lower than SAP Business Suite Enhancement Package 5, use the existing functions in that release. For more information, see Extraction of Supplier and Customer Master Data (Earlier ERP Releases) [external document]. In this case you need to configure the logical system for IDoc-XML. The steps are as follows:

1. Create an XML-file port
   
   Use transaction WE21 to create an XML-file port for IDoc processing. Ensure that you have network access from your local client to the directory configured in the XML-file port. Enter the port name, description, and the physical directory. Enter the function module EDI_PATH_CREATE_CLIENT_DOCNUM. On the Outbound Trigger tab enter the RFC destination LOCAL_EXEC.

2. Create Logical System
   
   Open transaction SALE and then go to Basic Settings Logical Systems Define Logical System to create a new logical system.

3. Maintain Distribution Model
   
   Open transaction SALE and then go to Modelling and Implementing Business Processes Maintain Distribution Model and Distribute Views. You can also use transaction BD64 for this.
   
   1. Switch to change mode and choose Create Model View to create a new entry. Enter a short text and a technical identifier.
   
   2. Choose Add Message Type for the newly created model. Enter a logical source system name and a destination system name and choose the message type DEBMDC.

4. Create Partner Profile
   
   Run transaction SALE and then go to Modelling and Implementing Business Processes Partner Profiles Generate Partner Profiles. Alternatively you can use transaction BD82.
   
   1. Select the newly created model using the input help for the technical name and then select the logical destination system.
   
   2. Enter the authorized user and the following values:
      
      ○ Version: 3
      ○ Pack.Size: 100
      ○ Output Mode: Immediate Transfer
      ○ Inbound. Processing: Immediately
3. Choose **Execute**. You can ignore the port error that appears.

5. Call transaction **WE20** and make the following settings:
   1. Open the **Partner Type LS** folder and select the partner profile you created above.
   2. Update the message type DEBMDM in the **Outbound Parameters** section. The Receiver Port is the XML-file port from the first step above. In the **Basic Type** field enter **DEBMDM06** for **DEBMDM**.

6. Test creation of IDOC XML
   1. Generate the IDoc-XML for customer using the transaction **BD12**.
   2. Check the newly generated IDocs using transaction **WE02** or **BD87**. You can use the receiver port as the filter criteria in the **Partner Port** field.
   3. Use transaction **AL11** to find the XML files on the directory of your XML-file port.
   4. To download the file for analysis purposes to local directory use transaction **CG3Y**.

### Data Cleansing in NetWeaver MDM

Data cleansing, that means matching, merging, validation, and enrichment of data can be done with SAP NetWeaver MDM. The files from the data extraction process have to be converted in a first step and then can be imported into SAP NetWeaver MDM.

After data cleansing steps the customer data can be extracted again in order to import them into the MDG Hub. Additionally key and value mapping information might be created and uploaded with the customer data.

To convert the data for consumption in NetWeaver MDM you can use the WebDynpro Application **MDG_TRANSFORMER_FPM_CMP** which can be called from the work center for MDG-C under **Data Exchange > Data Transfer > Convert Master Data**.

Use the transaction **FILE** to create the logical directory **MDG_TRANS_SOURCE** to be able to use the **File Converter**.

### Data Import into Target System (MDG Hub)

**Note**

To perform an initial load of master data, we recommend to use the functions described in the following documents:
- **Importing Master Data** [external document]
- **Data Transfer** [external document]

If you want to use parallel processing for data import it is recommended that you register your queue name prefixes in the QIN Scheduler so that they are executed automatically and do not have to be manually activated each time. To register the queue name follow these instructions:

1. Run transaction **SMQ2**.
2. Choose the **Execute**. This shows the list of current queues.
3. Choose **Goto > QIN Scheduler**.
4. Choose the **Registration**.
5. In the Queue Name field, enter a prefix for your queues. Add an asterisk after the name to allow you to add additional text to the prefix when scheduling the export or import. Enter additional details as required.
6. Press OK.

Your new queue has been registered and will execute automatically.

The IDoc-XML files contain the following fields in the header section:
To be able to import IDoc-XML files the following setup activities need to be carried out:

1. Use transaction \textit{IDX1} to create two ports in the IDoc adapter, one for sending and the other for receiving. Enter the port, client, description, and RFC destination for each port. Both ports should have the RFC destination of the MDG hub. Check that the port names match the names in your IDoc-XML file for the \textit{SNDPOR} and \textit{RCVPOR}, see table above for details.

2. In transaction \textit{WE21} enter the receiver XML port using the same name as in step 1 above. Enter the port name under the folder XML File, and enter a description and a physical directory. In the function module field enter \texttt{EDI\_PATH\_CREATE\_CLIENT\_DOCNUM}. On the \textit{Outbound: Trigger} tab, in the RFC destination field, enter \texttt{LOCAL\_EXEC}.

3. In transaction \textit{BD54} enter the sender and receiver partner numbers as logical system names.

4. In transaction \textit{FILE} create the logical file name. Enter a \textit{Logical File} and a \textit{Name}. In the Physical File field enter \texttt{<PARAM\_1>}. In the \textit{Application Area} field enter \texttt{CA}. In the \textit{Logical Path} field enter the logical file path.

5. In transaction \textit{AL11} make sure that the IDoc-XML files are stored under the logical path and that there are no other files stored in that directory. Double-click on the path to view the existing IDoc-XML file. You can use transaction \textit{CG3Z} to copy a local IDoc-XML file to the path.

6. To test the data import, open \textit{Data Exchange > Data transfer > Import Mater Data} in the \textit{SAP NetWeaver Portal} or \textit{SAP NetWeaver Business Client}. For more information, see \textit{Importing Master Data} [external document].

You can use the \textit{Monitor Data Transfer} [external document] application to get an overview of your data transfer processes.

### 3.23 Event Control

Check the table \textit{CRMC\_BUT\_CALL\_FU} and make sure that for the following entries the \textit{Call Function Module} indicator is not set:

- \texttt{COM\_BUPA\_MWX*}
- \texttt{BUPA\_OUTBOUND\_ALE\_MAIN}
- \texttt{BUPA\_OUTBOUND\_MAIN}

Check the table \textit{CRMC\_BUT\_CALL\_FU} and make sure that for the following entries the \textit{Call Function Module} indicator is set:

- \texttt{ABA\_FSBP\_INBOUND\_MAIN}
3.24 Validations and Enrichments

For an example of how to derivate the value of a field if you enter a specific value into another field, see Derivation of Fields in the Business Partner Data Model [external document].

3.25 Set Up Address Screening

The integration of SAP Fraud Management enables authorized users in change request processing in SAP MDG, central governance to perform the screening of a person or organization by SAP Fraud Management.

Prerequisites

- The integration of SAP Fraud Management is an optional feature in SAP MDG, central governance. To make use of it SAP Fraud Management 1.1 SP07 or higher has to be installed.
- The address screening features of SAP Fraud Management have to be enabled as described in the installation and upgrade guides on the SAP Help Portal under help.sap.com/fra in the section Enabling Address Screening Methods.
- Users of the address screening feature in the SAP Fraud Management system need to be assigned to the role SAP_BPCM_SYS_COM Business Partner Master Data Screening: System Communication:
  - You can assign this role to a service user that is specified in the definition of the RFC destination.
  - You can set up the RFC destination to require MDG users to log on to SAP Fraud Management as named users assigned to this role.
- You have set up an RFC destination of the Connection Type 3 (ABAP Connection) in transaction SM59. You can install SAP Fraud Management in the same SAP NetWeaver system as MDG, central governance. Nevertheless the RFC destination is required, in order to log on to the client in which SAP Fraud Management is running.
- In the Customizing activity under Master Data Governance, Central Governance > General Settings > Data Quality and Search > Validations and Enrichments > Define Enrichment Spots the entry ADDRESS(Screening) is available.

Note

The ACCESS_GROUP authorization check for fraud alerts in SAP Fraud Management is supported for MDG, business partner master data screening. If a change request from MDG specifies a value for ACCESS_GROUP, then this value is attached to any alerts created by the change request. Subsequent calls from MDG to change the status of the alert are checked against this value.

If you plan to use access groups to limit the set of alerts to which an MDG approver has access, then you should use the named-user approach to setting up the RFC communications. In this case, your named-users for approvers can have the access group authorizations that each approver requires. An authorization error is returned to MDG as an RFC error with accompanying message.
Activities

- To activate the address screening for SAP MDG, central governance, run the Customizing activity under Master Data Governance, Central Governance > General Settings > Process Modeling > Change Requests > Configure Properties of Change Request Step.
  1. Select the change request type and choose Change Request Step.
  2. Select a change request step and choose Enhancements and Checks per Change Request Step.
  3. Set the Relevant indicator for the entry Address Screening.

- In the client 000 in the Customizing activity under Master Data Governance, Central Governance > General Settings > Process Modeling > Workflow > Activate Event Type Linkage, the receiver types ACTIVATED_ACS and ROLLED_BACK_ACS are available. These receiver types have to be taken on the corresponding client and the Type Linkage indicator has to be set.

- To specify an RFC destination for the address screening run the Customizing activity under Master Data Governance, Central Governance > Master Data Governance for Business Partner > SAP Business Partner Screening > Specify RFC Destination for SAP Business Partner Screening.
  Use the entry help to select an RFC destination and save your settings.

Result

You have set up the address screening for MDG, central governance.

More Information

For information on SAP Fraud Management in general and on how to install and upgrade it, see the SAP Help Portal under help.sap.com/fra.
4 Configuring Master Data Governance for Customer on Client System

The business function *Master Data Governance for Customer on Client (ERP)* offers features that can be used as supplement to SAP Master Data Governance for Customer (MDG-C). It enables you to search and compare corresponding data on the hub system, before creating or changing customer master data on a client system. Additionally it enables you to trigger a copy of customer data from the hub system to the client system.

Prerequisites

**Master Data Governance for Customer**

*Master Data Governance for Customer on Client (ERP)* expects MDG-C to be configured in the hub system first and in the client system.

<i>Note</i>

For detailed information how to configure Master Data Governance for Customer see, *Configuring Master Data Governance for Customer.* [external document]

**Business Functions**

In the Customizing activity *Activate Business Functions* (transaction SFW5) you have activated the following business functions on the client system:

- Master Data Governance for Customer on Client (ERP) (MDG_CUST_ERPCLIENT_1) [external document]
- Business Foundation: Services for Supplier (CA_SUPPLIER_SOA) [external document]
- Business Foundation: Services for Business Partner (CA_BP_SOA) [external document]
- Improvements for SOA Reuse Functions (FND_SOA_REUSE_1) [external document]

**Process**

**Settings in the SOA Manager**

The configuration steps listed below have to be carried out in the SOA Manager (transaction SOAMANAGER) after the hub system has been configured as described in the document *Configuring the SOA Manager for Master Data Governance* [external document]:

- **Configure the search in the hub system**
  1. On the *Service Administration* tab choose *Business Scenario Configuration*.
  2. Choose *Create*, enter a Name and a Description and choose *Next*.
  3. Choose *Add*, enter the search term BusinessPartnerBasicDataByElementsQueryResponse_In in the SOA Management dialog box and choose *Go*.
  4. Select the entry and choose *Add To Worklist*. 
5. Select the entry, choose Assign Profiles, select MDG in the Assign Profile dialog box, choose Assign Profiles and choose Next.

6. Choose Add, enter the search term MDG_BS_BP_SEARCH and choose Go.

7. Select the entry and choose Add to Worklist.

   Note
   Make sure that the Communication Protocol is Use WS Protocol.

8. Select the entry and choose Assign Business Application.

9. Select a target system (client system) and choose Assign to Service Group.


   Note
   A dialog box is displayed, asking whether you want to activate the business scenario.
   We recommend not to activate the business scenario until the business scenario in the client system is activated.

- **Configure the search in the client system**
  
  To configure the search in the client system proceed exactly as you did for the hub system, except Step. 9. Here you have to select the hub system as target system.

   Note
   To activate the business scenario configuration choose Activate and then choose Process List on the Pending Tasks screen.

   Note
   To check whether the configuration works properly choose transaction XDO3 in the client system. On the Customer Display: Initial Screen dialog box use the input help of the Customer field. On the Customer Account Number dialog box select the Customer in MDG hub by Address Attributes tab and search for an existing business partner on the hub system.
   If necessary, deactivate the business scenarios in the hub system and in the client system, and activate them again in the same order.

- **Configure the client maintenance scenario**

   Note
   You can access this specific Customizing using the transaction DRFIMG.

   To define the technical settings for business systems, in the hub system run the following activity in Customizing for Data Replication under Define Custom Settings for Data Replication Define Technical Settings Define Technical Settings for Business Systems.

   1. Choose New Entries and enter the following data:
      - Business System <client system>

         Note
         The error message that might be displayed can be ignored.
2. Choose Enter, select your entry and choose Define Bus. Systems, BOs.
3. Choose New Entries and select the BO Type 147 Business Partner using the input help.
4. Choose Define Bus. Systems, BOs, Communication Channel, then New Entries and select the following entries:
   ○ Communication Channel Replication via Service
   ○ Key Harmonization Key Mapping

      Note
      In the Storage Location for Data Replication field you can decide whether changes are written into the staging area and change requests are created or whether changes are written into the active area without a workflow being triggered.

5. Choose Save.

      Note
      If you intend to maintain business partner relationships, repeat the procedure using BO Type 1405 Business Partner Relationship with the same settings.

      Note
      To check whether the configuration works properly proceed as follows:
      1. In the client system choose transaction XD01 and create a new customer.
      2. Make sure that the Customer-Vendor-Integration (CVI) creates the corresponding business partner.
      3. In the hub system check if a change request or a business partner has been created, depending on your selection in the Storage Location for Data Replication field.

**Configure the copy function**

To define the technical settings for business systems, in the client system run the following activity in Customizing under Master Data Governance, Central Governance General Settings Value Mapping Define Technical Settings for Business Systems.

1. Choose New Entries and enter the following data:
   ○ Business System <hub system>

      Note
      The error message that might be displayed can be ignored.

   ○ Logical System <hub system>
   ○ RFC Destination <hub system>
2. Choose Enter, select your entry and choose Define Bus. Systems, BOs.
3. Choose New Entries and select the BO Type 147 Business Partner using the input help.
4. Choose Save.
To check whether the configuration works properly, choose transaction XD03 in the client system as described above, and copy an existing business partner from the hub system to the client system.

To make the Copy button appear on the search result list, the authorization object MDGC_LCOPY has to be assigned to the user.

For further information, see SAP Note 1655835.
Adapting Master Data Governance for Customer

This documentation provides the information you require to change and enhance settings for Master Data Governance for Customer. It supplements the information provided in the section Configuring Master Data Governance for Customer.

5.1 Data Modeling

The purpose of data modeling is to define the structure of the data storage. During the master data processing, a change request is used that stores the master data changes in a staging area. The data model can define a reuse area that is used for data storage after the change request processing has been completed and the related data has been activated. In this case, the system moves data from the staging area to a storage location that is connected by the access class of the reuse area. This storage location is called active area.

If there is no reuse area defined, the same database tables that are used for the staging area, are also used to store active data. Then, no access class is involved, the system does not move data from one location to another, and MDG is used as the active area.

5.1.1 Concept of the MDG Data Modeling

A data model in Master Data Governance is comprised of various elements (entity types, attributes, and relationships) to enable you to model master data structures of any complexity in the system. These elements are described below.

The system uses the data model to generate database tables for storing the data you enter when managing the master data. Which key fields and non-key fields these database tables contain depends on the structure of your data model.

Note

This documentation supplements the information that is available in Customizing for Master Data Governance under General Settings > Data Modeling > Edit Data Model. Therefore, this documentation only covers the areas of the data model that require more background information for better comprehension.

Entity Types

You define entity types to model different types of master data in your data model. The system generates for each entity type the database tables that are needed for processing the master data in Master Data Governance. You can define the following properties for an entity type:

- You use the validity of entity field to specify for an entity type whether the validity of master data changes is restricted to editions:
edition

When you create a change request to process entities for entity types with this validity concept, you need to assign the new change request to an edition.
The corresponding change request type must reference an edition type.
With this validity concept, the edition field is included in the database tables.

no edition

When you create a change request (to process entities) for entity types with this validity concept, you do not assign the new change request to an edition unless you want to create edition-dependent hierarchies in the change request. The corresponding change request type does not have to reference an edition type in this case.
With this validity concept, the edition field is not included in the database tables.

You use storage/use type to specify whether and how master data can be changed in Master Data Governance. The storage and use type also indicates which database tables are generated by the system:

1 - changeable via change request: generated database tables

The master data of this storage and use type can be changed in Master Data Governance with a change request. The system generates all necessary database tables: check and text tables as well as additional tables, for example, for attachments and sets.
The common key fields of these tables are:

○ The entity type itself
○ The edition – if you previously specified in the data model that the validity of master data changes is restricted to editions
○ The entity types that are assigned to the entity type through leading relationships
Furthermore, all tables contain a checkbox that indicates whether the master data record is active. If the entity is stored in the MDG active area, this checkbox separates data in the staging from active data. If the entity is stored in a re-use area, this checkbox is used to mark a copy of the data from the active area as a snapshot at the time when the change request was created. This snapshot is used during activation of the data to detect conflicting changes that were done directly to the active area.
The settings you make for the entity type (such as language dependency) result in additional key fields in the text table and the tables for attachments and sets.
The non-key fields contained in the text table are the entity texts. The non-key fields contained in the check table are the attributes of the entity type. The attachment and set tables contain predefined non-key fields. Furthermore, all database tables contain a checkbox that indicates the deletion of the master data record. For entities that are edition-based, this checkbox indicates the end of validity in time. For entities that are stored in a re-use active area, this checkbox is considered during activation to check if data needs to be deleted. The check table also contains attributes that record which user created or changed the data records and when this was done.

2 - changeable w/o change request: generated check/text tables

The master data of this storage and use type can be changed in Master Data Governance without a change request. The system generates only the check and text tables with the entity type as well as with the entity types assigned to the entity type through leading relationships as fixed key fields.
The non-key fields contained in the text table are the entity texts. The check table does not contain non-key fields.

3 - not changeable via MDG: no generated tables

The master data of this storage and use type cannot be changed in Master Data Governance. Therefore, the system does not generate database tables. Instead, the system derives the available values from the domain that is assigned to the data element – either from the assigned value table or from the domain.
fixed values. Entity types with this storage and use type are typically used as reference values in qualifying relationships.

○ 4 - Changeable via Other Entity Type; Generated Database Tables

The master data of this storage and use type can be changed in Master Data Governance only with a change request of an entity type with storage and use type 1. The entity type needs to be in a relationship with the relationship type leading and assigned as the To-entity type to an entity type with storage and use type 1. The system generates the check table as described for storage and use type 1, but also generates the entity types that are assigned through qualifying relationships as key fields. The system does not generate a text table, attachments, or sets since entity texts are not allowed for entity types with this storage and use type.

The constraints described in the “Relationships” section apply when you assign a storage and use type to an entity type.

● If you want to model the hierarchy arrangement of entities with specific entity types during master data processing, you can allow a hierarchy structure for each entity type. By doing this, you define whether these hierarchies are version-dependent or cross-name, or whether a combination of these attributes is to apply to the hierarchies.

○ If you create version-dependent hierarchies, you can define different hierarchy versions in Customizing for Master Data Governance under General Settings > Process Modeling > Create Hierarchy Versions.

○ In cross-name hierarchies, the structure of the substructures – that is, the entities with their lower-level entities – is identical in all hierarchies.

You also define in the data model which entity type is to be used as an end node (without lower-level nodes) in the hierarchy, which entity type is to be used as the root node (hierarchy name) of the hierarchy, and which additional entity types are permitted for the hierarchy. By default, a hierarchy can contain one root node and multiple entities of the hierarchy-defining entity type. You can also assign other entity types with storage and use types 1, 2, and 3 to the hierarchy.

Note

Entities of the entity type that you define as the root node cannot be used as lower-level entities in the hierarchy. No relationships with relationship type Leading can be defined for the entity type itself.

You can also specify in the data model whether the system is to restrict the validity of the hierarchy for an entity type to editions.

The system automatically generates the database tables needed to store the hierarchies. They contain the following key fields:

○ The edition – if you previously specified in the data model that the validity of the hierarchy for this entity type is restricted to editions

○ A checkbox that indicates whether the master data record is active

○ The higher-level entity type

○ The higher-level entity

○ The lower-level entity type

○ The lower-level entity

○ The hierarchy version (if the hierarchy is version-dependent)

○ The hierarchy name (that is, the entity of the entity type you defined for the root node) if the hierarchy is not cross-name.
The table also contains a checkbox (as a non-key field) that indicates whether the respective master data record has been deleted, as well as a sequence number. This number specifies the sequence of the respective lower-level entities.

- To define the technical properties of an entity type or the texts for field labels and input help on the user interface for Master Data Governance, you can assign an existing data element to the entity types with storage and use type 1. You must assign a data element to entity types with storage and use types 2 and 3. This assignment is not permitted for entity types with storage and use type 4.

  Note
  If no data element is assigned to an entity type, you must use this entity type as a To-entity type in at least one relationship with the type Leading or Qualifying.

- By assigning the type of key assignment, you specify for new entities created for an entity type whether the processors are to specify the key or whether this is to be assigned internally by the system. You can also define whether processors are permitted to later change these keys.

  Note
  If you select a setting other than Key Cannot Be Changed; No Internal Key Assignment as the type of key assignment, the generated database tables change as follows:
  ○ Check table
    The table key contains a technical field instead of the entity type and its higher-level entity types. The entity type and its higher-level entity types are included in the attribute area of the database table.
  ○ Hierarchy table
    The hierarchy table does not change.
  ○ Other tables
    The key for the tables contains a technical field instead of the entity type and its higher-level entity types.
    The system generates a mapping table. The database table key contains a technical field for the mapping table. The attribute area contains the entity type and its higher-level entity types. If you select the Key Can Be Changed; Internal Key Assignment Possible setting as the type of key assignment, an additional attribute field to store the temporary key is added to the mapping table.
      ○ If the entity type is used as a From-entity type in relationships (see the “Relationships” section), the technical field replaces the entity type and its higher-level entity types in the corresponding tables.

  You also need to assign a number range object in this data model that the system can use to derive the required temporary keys when keys are assigned internally (before the entity is activated).

  - If you want to define texts for an entity type, you can specify the length of the short, medium, and long texts and you can specify whether these texts are to be stored as language-dependent in the database tables.
  - You can enter a more detailed description of the objects in the data model (entity type, attribute, or relationship). For entity types without data elements, the system also uses this description for display in master data maintenance.
  - For entity types with storage and use type 3 whose data elements reference a check table that, in turn, does not reference a text table, you can specify texts for the source field (short text, medium text, and long text). The system uses the text that can be stored in this check table field as the descriptive text of the entity type’s attributes (for example, when formatting the input help).
  - You can specify whether deletion of entity types by means of change requests is permitted.
**Relationships**

To map the structure of your master data management in the data model, you define relationships between the individual entity types. The relationship type determines whether one entity type (From-entity type) is at a higher level than another entity type (To-entity type) or whether it is to be copied as an attribute of the other entity type in the check table:

- **Relationship type Leading (P)**
  
  In this relationship type, the From-entity type is at a higher level than the To-entity type. This means the From-entity type is added as a key field to the database tables of the To-entity type, along with all assigned entity types with storage and use type 4.
  
  The From-entity type from a leading relationship to an entity type with storage and use type 4 is the respective entity type with which you can process the master data of the To-entity type with a change request.
  
  You **cannot** specify a data element.
  
  The cardinality 1:N is only permitted for this relationship type when no data element is specified.
  
  The cardinality 1:0 is also permitted if:
  
  - The To-entity type has storage and use type 1, no data element, and no additional relationships with the relationship type Leading.
  - The To-entity type has storage and use type 4 and no additional relationships with the relationship type Qualifying.

- **Relationship type Qualifying (Q)**
  
  This relationship type is only relevant for From-entity types with storage and use type 4. It is used to define additional key fields in the database tables. For entity types of this storage and use type, the system – by default – transfers only the key fields of the From-entity type from the leading relationship.
  
  Only the cardinality 1:N is permitted for this relationship type. You **cannot** specify a data element.

- **Relationship type Referencing (-)**
  
  This relationship type declares that the From-entity type is to be used as an attribute of the To-entity type.
  
  The description (name) of this type of relationship is inserted as a non-key field in the check table of the To-entity type.
  
  The cardinalities 1:N and 0:N are permitted for this relationship type. A data element can be specified in this case. However, this data element must have the same technical properties as the data element assigned to the From-entity type.
  
  As an alternative to a referencing relationship with a From-entity type with storage and use type 3, you can also define the attributes for the corresponding To-entity type directly. You configure this setting in Customizing for Master Data Governance under General Settings > Data Modeling > Edit Data Model in the subdialog Attributes.

  **Note**

  Whether you use a relationship with the relationship type Referencing to define attributes or whether you define these directly (where possible) depends on the settings configured in the data model and on the use type of the attributes. For example, if an attribute is a value to which a unit or currency is to be assigned, you must define this directly. In contrast, if an entity type with storage and use type 3 that you can use for attribute definition already exists, you should define a referencing relationship.

Which relationship types are permitted depends on the storage and use types of the entity types:
5.1.2 Transportation of Data Models to the Target System

You can transfer data models for Master Data Governance from your test system to your target system by means of transport requests.

**Process**

To transport an **active** version of a data model to the target system, proceed as follows:

1. In Customizing for **Master Data Governance**, choose **General Settings ➤ Data Modeling** and then the **Edit Data Model** activity.
2. To activate the data model again, select it and choose ✨ (Activate).
   
   A dialog box appears.
3. Specify the transport request that you want to use to transport the active data model and save your entries.

   The active data model is transported to the target system. Once in the target system, the data model is activated automatically. This can have the following effects on the generated database tables in which the entities are saved:
   - The generated database tables are generated again.

---

**Example**

For an example of the database tables the system generates for entity types with different properties defined in the data model, see *Example: Generated Database Tables* [external document].

---

<table>
<thead>
<tr>
<th>From-Entity Type</th>
<th>Storage and Use Type</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>R3 P</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>R3 P1 Q2</td>
</tr>
<tr>
<td>2</td>
<td>R P</td>
<td>P</td>
<td>-</td>
<td>-</td>
<td>R Q</td>
</tr>
<tr>
<td>3</td>
<td>R P</td>
<td>P</td>
<td>P4</td>
<td>R</td>
<td>Q</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td>P1</td>
</tr>
</tbody>
</table>

1 – Exactly one relationship with this relationship type must exist for the To-entity type
2 – At least one relationship with this relationship type must exist for the To-entity type
3 – Only possible if a data element is assigned to the From-entity type
4 – Only possible for check tables with multiple key fields

---

**Figure 1**

---

PUBLIC

© Copyright 2016 SAP SE or an SAP affiliate company.

All rights reserved.
○ The generated database tables are adjusted.
○ If the entity type was removed from the current data model, the generated database tables are deleted.

Note
If a deletion of the active data model is transported, the generated database tables are not deleted – with the exception of the hierarchy tables.

To transport an inactive version of a data model to the target system, proceed as follows:
1. In Customizing for Master Data Governance, choose General Settings > Data Modeling and then the Edit Data Model activity.
2. Choose Table View > Transport and specify the transport request with which you want to transport the inactive data model.
3. Select the data model and choose Process > Transport > Include in Request.

In the dialog box that appears, specify that all lower-level entries are to be transported and save your entries.

Note
You can activate the transported inactive data model in the target system.
To do this, in Customizing for Master Data Governance in the target system, choose General Settings > Data Modeling and then the Edit Data Model activity.
Select the data model and choose (Activate).

5.1.3 Defining Data Models in the Configuration Workbench

You can use this Web Dynpro application to define and activate a data model to map master data in the system, along with its properties and relationships. The system uses this data model to generate database tables in which the master data can be stored.

You can assign a reuse active area to a data model or to individual entity types of a data model. Then the inactive portion of master data for this data model is stored in the generated tables and the active portion is stored in the database tables specified in the reuse active area.

Note
You can also assign a reuse active area on the level of an entity type.

Prerequisites

You have created any customer-specific data elements you want to use for the entity types in the data model or for their attributes.

If you use entity types with internal key assignments, you can define prefixes for internal key assignment. You do this in Customizing for Master Data Governance under General Settings > Define Prefixes for Internal Key Assignment.
Features

Selecting Data Models or Creating New Ones
In the Configuration Workbench screen, you can select a data model for editing or you can create a new data model. By default, the system displays all data models that are available for processing. For each data model you can see whether an inactive version of the data model exists alongside the active version and whether that version differs from the active version.

Working with Data Models and their Entity Types
After you select a data model for editing or create a new data model in the Configuration Workbench screen, the Data Model screen opens.

In the Data Model screen, you can complete the following tasks:

- Edit data model details
- Create and customize entity types that belong to a data model.
- Check the validity of your settings using the Check button.
- Activate changes using the Activate button.
- Enable and disable entity types, attributes, and relationships

For more information, see Adapting Standard MDG Content to Your Business Needs

Data Model Details Panel
In the Data Model Details panel, you can edit the data model description and view details such as version, and activation status.

Entity Types Panel
You can select an entity type or create a new one in the Entity Types panel. You can edit settings for a selected or newly created entity type using the tab pages.

Entity Details Tab
Entity Details is divided into the following sections:

- General Details
  You must define a Storage and Use Type for the entity type. In addition, you can provide other data, such as a description and a data element.
- Hierarchies
  You can indicate whether hierarchies are allowed and what properties they are allowed to have. You can only allow a hierarchy to be set up for entity types with storage and use type 1.
- Key Assignment
  You can indicate how keys are assigned to the entity type.
- Enablement Status
  You can enable entity types that are relevant to your business and disable entity types that are irrelevant to your business.
- Reuse
  You can specify a reuse active area and references to elements of the data dictionary.
- Texts
You can specify the fields of the check tables that contain the texts for an entity type. This is only possible for entity types of storage and use type 3.

Attributes Tab
Here you define the attributes of each entity type in the data model. Attributes are mapped as non-key fields in the generated database tables of the entity type. You also need to assign an existing data element to each attribute. The data element determines the technical properties of the attribute as well as the field labels and the input help texts on the user interface. Attributes can be defined as required entry fields or as optional fields. You use a currency-supplying attribute or a unit-supplying attribute to assign a currency or unit of measure to the attribute.

Incoming Relationships and Outgoing Relationships Tabs
Relationships can be viewed from the perspective of each of the entity types that are involved. For example, the HAS_ADRE relationship between BP_HEADER and ADDRESS can be viewed from the perspective of both entity types.

You can view the relationship in the following ways:
- If you select the BP_HEADER entity type, you can view the relationship in the Outgoing Relationships tab page.
- If you select the ADDRESS entity type, you can view the relationship in the Incoming Relationships tab page.

For all relationships, you can define properties such as:
- **Relationship Type** (leading, referencing, qualifying, or foreign key)
- **Cardinality**
- **Fields of foreign key relationships**

You can assign the key fields of the from-entity type to the attributes and key fields of the to-entity type.

Example
In the PFLI entity type of the SF data model, you model flight scheduling data. For example, you can specify the cities CITYFROM and CITYTO. The GEOCITY entity type has a storage and use type of 3. It acts as a check table for valid cities. If you want to ensure only valid cities are selectable, you create a foreign key relationships between CITYFROM and GEOCITY, and between CITYTO and GEOCITY.

To maintain the foreign key attributes for PFLI, you can open the **Incoming Relationships** tab, select the relationships CITYFROM and CITYTO, and choose the foreign keys button. You want to define foreign key relationships so that the fields PARTNER_1 and PARTNER_2 at entity type BPREL contain only the values of the field BP_HEADER at entity type BP_HEADER.

Business Object Types Tab
You have to assign business object types only for entity types of storage and use type 1 that you want to replicate, or for which you want to generate their own Enterprise Search template.

If you have assigned the same business object type to multiple entity types, then you have to specify the entity type to be used for each business object type.

You can do this in Customizing for **Master Data Governance** under **Data Modelling > Specify the Entity Type to Be Used for Each Business Object Type**.

Hierarchies Tab
If you want it to be possible to set up a hierarchy for the entity type, you must specify at least the root node (hierarchy name) for the hierarchy here. To do this, choose one of the available entity types and assign **Hierarchy**.
Name as the usage type. You also can specify all entity types that are to be allowed in the hierarchy of the entity type (No Special Use or Ranges Permitted on End Nodes)

5.2 UI Modeling

The purpose of UI modeling is to define and customize user interfaces with which users process master data.

5.2.1 Managing of UI Configurations

You use the Manage UI Configurations (USMD_UI_CONFIGURATION) Web Dynpro application to manage user interfaces in SAP Master Data Governance. Each table row represents a separate user interface and consists of the user interface application and its configuration. You can create a new user interface configuration by copying an existing one. You can also edit the configurations for existing user interfaces. Each link you click opens the relevant screen in the Floorplan Manager (FPM).

Note
You can only use this function if Business Function Master Data Governance, Generic Functions 7.0 Feature Pack (MDG_FOUNDATION_5) is active.

The previous version of this application only allows management of UI configurations for specific types of single-object processing UIs.

If the relevant business function is not active, you can edit the relevant technical elements using transaction SE80. For more information, see the links in this document under Activities Working with a UI Configuration. The documents listed cover editing using transaction SE80 as well as editing using this Web Dynpro application.

The most common types of user interface that you can manage are as follows:

- Single-Object Processing
- Multiple-Record Processing
- Search

There are many options to change a user interface including customizing, enhancement, context-based adaptation (CBA), and personalization. Some options affect all clients of a system. Other options are client specific. It is even possible to restrict changes to only one user. For more information, see Floorplan Manager for Web Dynpro ABAP [external document].

Prerequisites

- An active data model exists.
- You have basic knowledge of how to use the FPM and of the configuration of applications and components with Web Dynpro ABAP.
- To create a new user interface by copying an existing one, the following criteria must be met:
  - You can use an active MDG data model with at least one entity type with storage and use type 1.
You have assigned a business object type code (OTC) to this entity type.

Before starting the configuration you need to carry out the following steps to ensure the default data model as the data model for which the UI is configured in the following way:

1. Run transaction `SPERS_MAINT`.
2. Select `Edit Objects`.
3. From the displayed list, choose `SAP Master Data Governance - R_FMDM_MODEL`.
4. In the pop-up, set the value of the field `Standard Data Model` to the model that you want to use for UI processing.
5. Confirm and save.

Activities

Opening the Web Dynpro Activity in Customizing

Path in Customizing for `Master Data Governance, Central Governance` (transaction `MDGIMG`):

- General Settings
- UI Modeling
- Manage UI Configurations

Copying a User Interface Configuration

1. Select the UI configuration you want to copy and choose the `Copy` button.
2. To expand configurable components, choose the `Configurable Components` button.
3. In the `Copy` column, select the technical elements you want to copy, and enter appropriate names for the target configurations.
4. Choose the `Start Deep-Copy` button.
5. Return to the `Manage UI Configurations` screen and refresh the table content. The system displays an additional row in the table with the configurations you just created.
6. If the user interface is compatible with the `MDG Communicator`, the `MDG Communicator Status` is set to `Configuration missing`. To make the `MDG Communicator` available, you must configure it by choosing the `Details` link.

Subsequent steps depend on the type of user interface you are configuring and the type of configuration you want.

Working with a UI Configuration

The following documents provide detailed information on the concept behind the particular types of user interfaces, and instructions on how to create new user interfaces either using the Web Dynpro application `USMD_UI_CONFIGURATION` or using transaction `SE80`:

Single-Object Processing

- Concept: `Creating User Interfaces for Single Object Processing` [external document]
- Instructions: `Creating a Basic Configuration for the Single-Object Processing UI` [external document]

Search

- Concept: `Configuration of the Generic Search` [external document]
- Instructions: `Configuring the Generic Search for a Particular Business Object Type` [external document]
5.2.2 Context-Based Adaptation for Business Partner

With context-based adaptations, FPM applications can be adapted dependent on specific runtime parameters. A particular adaptation is applied when the application is called within a specified context.

Features

SAP Master Data Governance for business partner comes up with several context-based adaptations in its standard delivery, such as:

- In the business partner creation dialog a certain amount of fields is offered depending whether the business partner to be created is a person or an organization, for example date of birth is offered for persons only.
- In the address maintenance dialog certain fields are offered and arranged to fit the country specific manner to represent addresses.

More Information

For further details on context-based adaptation, see help.sap.com/saphelp_nw73ehp1/helpdata/en/f5/823cdb32254707aec436de005e0ada/content.htm

5.2.3 Configuring Business Context Viewer for MDG Customer

You can use this function to view context-related information for your customer master data in a side panel. You must activate the Business Context Viewer (BCV) to access the side panels for the Web Dynpro applications Create Customer and Change Customer.

Prerequisites

1. To enable BCV, you must activate the following business functions:
   - FND, Business Context Viewer Main Application (/BCV/MAIN)
   - FND, Business Context Viewer Main Application 2 (/BCV/MAIN_1)
   - FND, Business Context Viewer NWBC Side Panel (/BCV/NWBC_SIDEPANEL)

2. The business content for MDG Customer is delivered in the BC sets BCV Content for MDG Framework (MDGAF_BCV) and BCV Content for MDG BP Release 7.0 (MDGBP_BCV_70). You must activate them in transaction SCPR20.
Process

To view this content, open the BCV side panel by choosing the Side Panel link in the upper right corner of your MDG Customer user interface. From the side panel, select the following overviews you require from the dropdown list:

Sales Overview
This BCV content provides you with a list of all sales orders created for the current customer number.

Changes Overview
This BCV content provides you with a list of changes raised by the current MDG change request.

More Information

For more information about BCV, see Business Context Viewer (BCV) [external document]

5.3 Data Quality and Search

The data quality functions of MDG allow you to enrich and validate master data, as well as to prevent the creation of duplicates. The various search capabilities are not only used to find master data that can be processed, but are also used for matching data to prevent the creation of duplicate information. Correct and complete data can be achieved with automatic derivation of attributes and enrichment from external data sources.

5.3.1 Search Providers for Master Data Governance

In SAP Master Data Governance you can use the following search providers to search for master data:

- Enterprise Search [page 71]
- Database Search [page 73]
- Business Address Services (BAS)-Based Search [page 74]
- Searching with Customer-Specific Search Providers [page 75]
- SAP HANA Search

Note

To configure SAP HANA Search see Configuring SAP HANA-Based Search for MDG [page 42] and Configuring Drill-Down Search (Optional) [external document].

5.3.1.1 Enterprise Search

In SAP Master Data Governance you can use Enterprise Search to search for master data in the staging area and the active area.
Features

The standard system contains the following default search application and access class in Customizing for Master Data Governance, Central Governance under » SAP Customizing Implementation Guide » Cross-Application Components » Processes and Tools for Enterprise Applications » Master Data Governance, Central Governance » General Settings » Data Quality and Search » Define Search Applications:

Table 27

<table>
<thead>
<tr>
<th>Search Application</th>
<th>Access Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>ES</td>
<td>CL_SDQ_USMD_SEARCH_DATA_IMPL</td>
</tr>
</tbody>
</table>

The following additional search options are available for Enterprise Search:

- Free text search
- Fuzzy search

When you select the respective checkboxes, two additional fields appear on the Enterprise Search user interface. In the standard system these two checkboxes are selected for search application ES.

Activities

Note

To use Enterprise Search for SAP Master Data Governance, step one and step two of the procedure must be executed only if you use customer-defined objects. Step three and step four must always be executed.

1. Search object connector template:

   The standard system contains default search object connector templates for Supplier and Material. However, if you want to use customer-defined objects, the following is required:

   For every business object type you want to search for, you have to create a search object connector template in Customizing for Master Data Governance, Central Governance under » SAP Customizing Implementation Guide » Cross-Application Components » Master Data Governance, Central Governance » General Settings » Data Quality and Search » Create Search Object Connector Templates.

2. You then assign your search object connector templates to your business object types in Customizing activity Assign Search Object Connector Templates to Object Types.

   The standard system contains the following default assignments:

   Table 28

<table>
<thead>
<tr>
<th>Business Object Type</th>
<th>Search Object Connector Template</th>
</tr>
</thead>
<tbody>
<tr>
<td>986 (Business Partner Relationship Process Control)</td>
<td>MDG_BUSINESS_PARTNER</td>
</tr>
<tr>
<td>194 (Material)</td>
<td>MDG_MATERIAL</td>
</tr>
</tbody>
</table>

3. Search object connector:

   In the Connector Administration Cockpit (transaction ESH_COCKPIT) you must create a search object connector based on the respective search object connector template and software component.
Recommendation
On the detailed screen for the search object connector template, go to the Schedules tab page and enable real-time indexing.

4. Search index:
   In the Connector Administration Cockpit you have to execute the following:
   - Initial data load for indexing the data for your search object connector(s).
   - Schedule the delta update of the search index so that the Enterprise Search is regularly updated with the master data that has been changed in the meantime.

The concept of Enterprise Search for SAP Master Data Governance described above is shown in the following figure:

![Diagram of Enterprise Search for SAP Master Data Governance](image)

Figure 2: Enterprise Search for SAP Master Data Governance

5.3.1.2 Database Search

In SAP Master Data Governance you can use the database search to find master data for changing or verification. It is an exact search method that is based on exact values or value ranges like identification numbers or names that are stored in databases.
5.3.1.3 Business Address Services-Based Search

In SAP Master Data Governance you can use the Business Address Services-based (BAS-based) search to find master data for changing or verification. It is a fuzzy search method that takes misspelled or similar values into account.

Prerequisites

If you want to use the BAS-based search, you require a data quality management solution, for example SAP BusinessObjects Data Quality Management (version for SAP Solutions).

Features

The standard system contains the following default search application and access class in Customizing for Master Data Governance under SAP Customizing Implementation Guide > Cross-Application Components > Master Data Governance > General Settings > Data Quality and Search > Define Search Applications:

Table 29

<table>
<thead>
<tr>
<th>Search Application</th>
<th>Access Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>AD</td>
<td>CL_MDG_BS_ADR_SEARCH</td>
</tr>
</tbody>
</table>

The concept of the BAS-based search for SAP Master Data Governance is shown in the following figure:

Figure 3: BAS-Based Search for SAP Master Data Governance
More Information

For more information about Business Address Services (BAS), see Business Address Services (BC-SRV-ADR) [external document]
For more information about SAP BusinessObjects Data Quality Management, see SAP Library for SAP BusinessObjects on SAP Help Portal at help.sap.com:

5.3.1.4 Searching with Customer-Specific Search Providers

In SAP Master Data Governance you can also implement your own search providers. If you want to do this, you have to do the following:

Procedure

Mandatory settings for search processing

1. In Customizing for Master Data Governance, enter your specific settings under [SAP Customizing Implementation Guide > Cross-Application Components > Master Data Governance > General Settings > Data Quality and Search > Define Search Applications]
   ○ Define your search application.
   ○ Define your access class.

   Note
   Your access class must use the standard search interface IF_USMD_SEARCH_DATA (Search for Entities).

2. User interface: Use the generic WebDynpro application USMD_ENTITY_SEARCH and launch it with the parameter SEARCH_MODE = your new search application (as defined in step 1).

Optional search indexing

1. Initial load of index: Use the class CL_USMD_MODEL_EXT to read or extract data from the Master Data Governance data models.
2. Delta load of index: Use the enhancement spot USMD_TRANSACTION_EVENTS to update the index with the changes made in the records of a Master Data Governance data model.

5.3.1.5 Configuring SAP HANA-Based Search for MDG

SAP HANA-based search for SAP Master Data Governance enables you to perform searches and duplicate checks on master data residing in the SAP HANA database. An SAP HANA search provider is delivered to enable these features.

The following data models are supported out-of-the-box for MDG on HANA:

• Flex data models
- The business partner reuse model (BP)
- The material reuse model (MR)

The access class implementation is not provided for other reuse models. You must implement the access class for SAP HANA search to use it with the other reuse models.

SAP HANA-based search for SAP Master Data Governance can be used for the following MDG applications:
- Master Data Governance for Custom Objects
- Master Data Governance for Financials
- Master Data Governance for Supplier
- Master Data Governance for Customer
- Master Data Governance for Material

Prerequisites

You have activated the business functions Master Data Governance, Generic Functions 7.0 or higher (MDG_FOUNDATION_4 or higher) and Master Data Governance, Generic Functions 7.0 Feature Pack or higher (MDG_FOUNDATION_5 or higher).

You have installed the SAP HANA database, support package 06 or higher. We recommend that you install the highest available version of the SAP HANA database.

You must have the following permissions to work with search views in SAP HANA:
- Permission to create a package and to write objects into packages
- Permission to create, change and drop attribute views
- Permission to create, change and drop SQL views
- Permission to create, execute and drop rule sets

For more details please refer to the SAP HANA security guide.

Process

To configure SAP HANA-based search for MDG, carry out the steps described below.

1. Create Database Connection

Run transaction DBCO and create a database connection to the SAP HANA database.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Connection Name</td>
<td>Unique name for the SAP HANA database connection used for search and duplicate check</td>
</tr>
<tr>
<td>Database System</td>
<td>SAP HANA database</td>
</tr>
<tr>
<td>Permanent</td>
<td>Yes</td>
</tr>
<tr>
<td>User Name</td>
<td>Schema name created in step above</td>
</tr>
<tr>
<td>Connection Information</td>
<td>Server: instance number</td>
</tr>
<tr>
<td>Connection Limit</td>
<td>0</td>
</tr>
</tbody>
</table>
### Optimum Number of Connections

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimum Number of Connections</td>
<td>0</td>
</tr>
</tbody>
</table>

### Note
#### Deployment Options for MDG

MDG can be deployed on an SAP HANA database or on any database.

- If you deploy MDG on SAP HANA, then SAP HANA acts as the primary database and the creation of the database connection is optional. If the database connection is not maintained then a default connection is derived automatically.
- If you deploy MDG on any other database, then you must maintain the database connection to the schema in the SAP HANA database.

### 2. Maintain the MDG SAP HANA Database Profile Settings

You must define the MDG landscape settings, such as the connection to the SAP HANA database that is used for the search and duplicate check processes. You can make these setting in Customizing under [Master Data Governance, Central Governance] > [General Settings] > [Technical Settings for Master Data] > [Define MDG Landscape Profile].

The use of an SAP Landscape Transformation (SLT) server is optional for MDG data replication. If you use SLT for replicating the MDG table data to the SAP HANA database system, then you must also define a connection to an SLT server as explained below.

### Note
#### Deployment Options for MDG

- If you deploy MDG on SAP HANA, then SAP HANA acts as the primary database and no replication is required.
- If you deploy MDG on another database, the MDG data must be replicated to SAP HANA search schema either by SAP Landscape Transformation (SLT) or by other means. To generate a search view in the target system where search is performed, the MDG table metadata and data must be replicated to the SAP HANA database. To enable this initial replication of the data you must carry out the steps described below.

1. Run transaction MDGIMG.
2. Navigate to [Master Data Governance, Central Governance] > [General Settings] > [Technical Settings for Master Data] > [Define MDG Landscape Profile].
3. Enter data in the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Connection Name for MDG</td>
<td>The SAP HANA database used for the search and duplicate check processes created in the previous step. This field is optional if MDG is deployed on a SAP HANA database.</td>
</tr>
<tr>
<td>RFC Connecting MDG to SLT System</td>
<td>Optional, only enter data if you use SLT for data replication</td>
</tr>
</tbody>
</table>
In the **SAP HANA** system, where the search on MDG data is performed, you must generate the search view. If you deploy MDG on a traditional database, and use **SLT** for replication then, when generating the view, before it is created, the system replicates the required table metadata to the **SAP HANA** database using the **SLT** settings.

If **SAP HANA** is the primary database, it is not mandatory to maintain the database connection name in MDG Landscape Profile customizing. If the name is not maintained the system uses the default database connection. You still have the option of maintaining a different connection name in the MDG Landscape Profile if you do not wish to use the default database connection.

### 4. In the SLT system the SLT user requires the authorization object **S_DMIS**, with the following field values defined for their role:

<table>
<thead>
<tr>
<th>Authorization Object</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity (ACTVT)</td>
<td>02 (Change)</td>
</tr>
<tr>
<td>MBT PCL: Scenario (MBT_PR_ARE)</td>
<td>SLOP (SAP Landscape Transformation)</td>
</tr>
<tr>
<td>MBT PCL: Processing Role Level (MBT_PR_LEV)</td>
<td>PACKAGE (Transfer package level)</td>
</tr>
</tbody>
</table>

### 5. For Material Search, in transaction **SA38** execute the report **MDG_HDB_MAT_MIGRATE_LONGTEXT** as a background job. Select the **Overwrite target table records** checkbox, to perform the initial load of material long texts to the database table **MDGHDB_LONGTEXT**. This loads the following long text types: Basic Data Text, Sales Text, Purchase Order Text, Inspection Text, Internal Note, and Material Note MRP.

### 3. Define Authorization Relevance for Each Entity Type

The authorizations maintained in customizing are considered during search. You can maintain the authorization in Customizing under **Master Data Governance, Central Governance > General Settings > Data Modeling > Define Authorization Relevance per Entity Type**.

### 4. Create Search View

Create a search view in the development system and transport it to the test and production systems. The search view must be generated or regenerated in the target (test and production) systems.

If you are using the business partner, customer, or supplier domains and have activated the business functions **MDG_ERP_CUSTOMER_3** or higher, **MDG_ERP_SUPPLIER_4** or higher, or **MDG_BUPA_1** or higher, or if you are using material domain and have activated the business function **MDG_MATERIAL_5** or higher, then you must assign the template views from these business functions to a customer package in the **Create Search View** configuration activity before you can generate and use them.

You must also have authorization to create a workbench request.

To create a search view, run transaction **MDG_HDB_GEN_UI** or navigate to **Master Data Governance, Central Governance > General Settings > Data Quality and Search > Search and Duplicate Check > Create Search View**.

The package where you generate the search view must be in the customer namespace. Enter the name of the package during search view creation.

When you create the search view and the system generates the **SAP HANA** view, the following search configuration data is automatically updated:
In SAP HANA attribute views are created on the active and inactive areas. After you create the search view it can be manually edited within SAP HANA Studio to update the search properties of an attribute. In this case, if the search view is regenerated, the new search view will overwrite the manually updated search view.

You can create a search rule set during the search view generation if you want the search to be performed based on search rule sets. If you choose the create ruleset option for a reuse model, a union SQL view is created on the attribute view in SAP HANA. This search rule set can also be manually updated according to the business requirements of the users after it is generated. If the search view is edited at a later date and is regenerated, the search rule set will not be regenerated/overwritten; it has to be manually adjusted.

You must manually check out the generated search rule set to the Project Explorer view of the SAP HANA Studio Administration Console before it can be edited to change any parameter, such as the fuzzy value or weight of an attribute, and activate it to enable search based on this modified search rule set.

You can also copy an existing search view and edit it before generating the search view.

If there is a mismatch between the generated search view and the underlying objects, the system recognizes this and updates the status of the generated search view to Outdated. You can edit this outdated search view and regenerate the view.

To delete a search view, you must first remove the customizing settings for the search view, and then delete the search view. The status of the view is then set to Marked for Deletion. In transaction SE38 execute the report program MDG_HDB_DELETE_SEARCH_VIEWS to delete the specific view or all views that are marked for deletion, and drop the corresponding objects in SAP HANA.

You must set filters in the SAP HANA staging views to exclude records that have the obsolete indicator set. Identify all the Obsolete Indicator flags. The fields corresponding to the obsolete indicator flags in each table of a staging view have the technical naming convention USMD*_OBS_* or USMD*_O_. Select the obsolete indicator in the Details section of the staging view, right click and select Apply Filter. In the Operator field select Not Equal and in the Value field enter X.

Table 33

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Operator</th>
<th>Filter Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>USMD<em><em>OBS</em></em></td>
<td>Not Equal</td>
<td>X</td>
</tr>
<tr>
<td>USMD*<em>O</em></td>
<td>Not Equal</td>
<td>X</td>
</tr>
</tbody>
</table>

For material search you must set filters in the SAP HANA views for the material-related long texts stored in the database table MUGHDB_LONGTEXT. This means that only the appropriate long texts are taken from MUGHDB_LONGTEXT. To do this, in the SAP HANA studio, open the Content folder and navigate to the package where the search views are created. For reuse entity types, creating a search view generates two views in the SAP HANA system (one each for the active and staging areas), or three if you are using classification data. The views generated for the active area have names similar to searchviewname_Reuse and searchviewname_RINOB.

Open the reuse SAP HANA views below. Go to Detail window, and select the long text table with the alias you want to update and right-click on the attribute. From the menu choose Apply Filter. From the drop-down menu choose the operator Equal and maintain the values as specified in the tables below.

Basic Text
Verify Customizing Settings for Search View

5. **Verify Customizing Settings for Search View**

After you have created and saved the search view, you must verify that the customizing settings are automatically updated for the newly created search view. To do this, perform the following:

1. Run transaction MDGIMG.
2. Navigate to Master Data Governance, Central Governance > General Settings > Data Quality and Search > Search and Duplicate Check > Define Search Applications.
3. Select the row with the **Search Mode** [HA (HANA)]. Note that the **Fuzzy** checkbox has no impact on SAP HANA search; SAP HANA search is fuzzy by default and this cannot be changed here.

4. Double-click on **Allocation of Search Help to Search Applications**.

5. Verify that there is an entry for the newly created search view in the **Included Search Help** field with the technical name provided during search view creation.

6. Select the row of the newly created search view.

7. Double-click on **Allocation of Entities to Search Help** and verify that the main entity type that you selected during search view creation is updated.

6. **Create Match Profile for Duplicate Checks based on SAP HANA Search**

   If you have created a search rule set in the Create Search View step, you can use it to configure the match profile for duplicate checks.

   1. Run transaction MDGIMG.
   2. Navigate to **Master Data Governance, Central Governance > General Settings > Data Quality and Search > Search and Duplicate Check > Define Search Applications**.
   3. Select the row with the **Search Mode** [HA (HANA)].
   4. Double-click on **Match Profile**.
   5. For the specific data model and the **Match Profile ID for Duplicate Check** enter the name of the search rule set if you generated one in step 4 above, otherwise, leave the field empty.

   When you enter the search rule set name, the information from the search rule set is used instead of the attribute view while performing search during duplicate checks.

7. **Configure Duplicate Check Based on SAP HANA search**

   After you have maintained a match profile ID, you can configure the search view for duplicate checks.

   1. Execute transaction MDGIMG.
   2. Navigate to **Master Data Governance, Central Governance > General Settings > Data Quality and Search > Search and Duplicate Check > Configure Duplicate Check for Entity Types**.
   3. Select the **Data Model** and **Entity Type** for which you want to configure the duplicate check. Select the **Search Mode** as [HA]. Enter the threshold values for the duplicate check. Enter the name of the **Match Profile ID** and the search view to be used for the duplicate check. Select the **Match Profile Based UI** if required.

8. **Test Search and Duplicate Check**

   1. Test the SAP HANA Search
      1. Launch the **SAP NetWeaver Business Client**.
      2. Select the work center for your data model.
      3. Launch the search UI.
      4. In the **Search Method** field enter the **SAP HANA Search Configuration** that you have created.
      5. Choose **Search** and the search results should be returned.
      6. In addition, perform a freestyle search and an attribute search and check the results.

   2. Test the Duplicate Check
      1. Create a duplicate of an existing object.
      2. When you have entered data for your object choose **Check**. This triggers the duplicate check and the system should indicate that your new object is a potential duplicate.
Result
You have now configured your system to use SAP HANA for MDG search. For drill down search configuration, see Configuring Drill-Down Search (Optional) [external document].

5.3.2 Definition of Validations and Derivations in BRFplus

You use this Web Dynpro application (USMD_RULE) to define validation and derivation rules for a specific data model of Master Data Governance. Derivation rules are designed to simplify the entry of master data, whereas validation rules ensure consistency of the master data entered.

Integration
After you specify a data model, the Web Dynpro application launches the Business Rule Framework plus (BRFplus) application. For more information, see Business Rule Framework plus [external document] (available in English only).

Prerequisites
You have created the data model for which you want to define derivations and validations.

Features

Trigger Functions
You can assign trigger functions to the following events to validate master data changes:

- **CHECK_ENTITY** (Check entity): The system executes the function assigned to this event when entity values are validated during single or collective processing, when mass changes are made, or during the upload process.
- **CHECK_CHANGE_REQUEST** (Check change request): The system executes the function assigned to this event when the change request is checked. It also executes the function prior to the final check of the change request.
- **CHECK_EDITION** (Check edition): The system executes the function assigned to this event before an edition is released.
- **CHECK_ENTITY_HRY** (Check hierarchy information of an entity): The system executes the function assigned to this event after the hierarchy of an entity type is changed.
- **CHECK_CHANGE_REQUEST_HRY** (Check hierarchy information of a change request): The system executes the function assigned to this event at the same time as CHECK_CHANGE_REQUEST. The function assigned to the CHECK_CHANGE_REQUEST_HRY event is used to check the hierarchy information.
- **CHECK_EDITION_HRY** (Check hierarchy information of an edition): The system executes the function assigned to this event at the same time as CHECK_EDITION. The function assigned to the CHECK_EDITION_HRY event is used to check the hierarchy information.
• **DERIVATION** (Derivation): The system executes the function assigned to this event when data is entered during single processing or the upload process.

When defining rules for **CHECK** events, you can define an action for issuing a message and control further processing in Master Data Governance based on which message type you select.

When defining rules for the **DERIVATION** event, you can use the *Change Context* action to change the attribute values of the entity type.

You can also trigger other actions (for example, by means of a static method) and include additional information in the computation.

For each entity type of the data model with storage and use type 1 or 4, the system generates a structure-type data object when you start this Web Dynpro application for the first time. Each time you subsequently activate a change to the data model, the system again generates the data objects the next time the user starts the Web Dynpro application.

**BRFplus Structure Generation**

If entity type 1 and entity type 4 have the cardinality 1:N, the BRFplus catalog generation does not add the fields of entity type 4 to the fields of entity type 1.

If entity type 1 and entity type 4 have the cardinality 1:1, the structure generation and the derivation function in BRFplus treat entity type 4 as an extension of entity type 1.

If entity type 1 and entity type 4 have the cardinality 1:1, the BRFplus catalog generation adds the fields of entity type 4 to the fields of entity type 1.

**Note**

If entity type 1 and entity type 4 have the cardinality 1:1, instead of calling the derivation function of entity type 4, the system calls the derivation function of entity type 1.

**Naming Conventions**

The following naming conventions apply to the relevant nodes, objects, applications, and catalogs:

- **Trigger function nodes in the catalog structure**
  
  The naming convention for validation-based trigger function nodes of a catalog structure is: **CHECK_<name of entity type>**, for example, **CHECK_ACCOUNT**.

  The naming convention for derivation-based trigger function nodes of a catalog structure is **DERIVE_<name of entity type>**, for example, **DERIVE_ACCOUNT**.

  **Note**

  These naming conventions apply only to the naming of the trigger function nodes of a catalog structure. They do not apply to the naming of the trigger functions linked to the nodes. Each entity type should have a maximum of one trigger function per event. The nodes of the respective function branch of the trigger function represent the corresponding application options.

- **Data objects**
  
  Data objects are automatically generated from the data model definition. The structure-type data objects have the same names as their respective entity types.

  **Note**

  The naming conventions valid for SAP enhancement package 4 are still supported. However, we recommend that you **discontinue** their use.
The standard system features the following predefined data objects, which you can use within rules and functions, as additional input parameters:

- `SAPFMDM_CREQUEST_TYPE`: Change of change request type ID
- `SAPFMDM_CREQUEST`: Change of change request ID
- `SAPFMDM_EDITION_TYPE`: Edition type ID
- `SAPFMDM_EDITION`: Edition ID
- `SAPFMDM_PROCESS`: Business activity
- `SAPFMDM_CREQUEST_STEP`: Change request step
- `SAPFMDM_CREQUEST_INDEX`: Change request index
- `SAPFMDM_WORKITEM_ID`: Work item
- `SAPFMDM_HRY_DELTA`: Deep structure consisting of a hierarchy relationship (or an associated pair of objects) for validation

This must be used in the hierarchy-based validation events `CHECK_ENTITY_HRY`, `CHECK_CREQUEST_HRY`, and `CHECK_EDITION_HRY`.

**Note**
If a trigger function contains the predefined data objects only, it is executed once during the validation.

- **BRFplus application and catalog**
  The system creates the BRFplus application [external document] and catalog structure automatically for each data model when you define validations or derivations.

**Note**
Use the namespace for the system-generated BRFplus application and catalog structure to create your own applications and catalogs. The system uses the syntax `FMDM_MODEL_<name of data model>`, for example, `FMDM_MODEL_OG` for data model OG.

### Enhancement for Validation and Derivation

The Business Add-In (BAdI) `Define Validations/Derivations` is provided as an enhancement implementation of this Web Dynpro application. This enables you to define validations and derivations using alternative code.

To call the BAdI, in Customizing for **Master Data Governance, Central Governance** choose > General Settings > Data Quality and Search and choose the Customizing activity `BAdI: Define Validations/Derivations`.

If entity type 1 and entity type 4 have the cardinality 1:N, in the derivation function of the entity type 4 you have read access to the values of entity type 1 for determining entity type 4.

In the derivation function of entity type 1, you do not have read access to the values of entity type 4.

### Activities

To start this Web Dynpro application, in Customizing for **Master Data Governance, Central Governance** choose > General Settings > Data Quality and Search and choose the Customizing activity `Define Validation and Derivation Rules`. 
Example

Creating BRFplus functions for derivations of entities with storage and use type 4

The following table shows an example of the different cardinalities, which are described in the BRFplus Structure Generation section.

Table 40

<table>
<thead>
<tr>
<th>Entities</th>
<th>Entity Type</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CARR</td>
<td>1</td>
<td>CURRCODE</td>
</tr>
<tr>
<td>CARR_DELTA</td>
<td>4 with 1:1 cardinality</td>
<td>URL</td>
</tr>
<tr>
<td>COUNTER</td>
<td>4 with 1:N cardinality</td>
<td>COUNTNUM</td>
</tr>
<tr>
<td>AIRPORT</td>
<td></td>
<td>C_URL</td>
</tr>
</tbody>
</table>

This example contains the entity CARR with storage and use type 1 along with the dependent entities CARR_DELTA and COUNTER, which have storage and use type 4. CARR_DELTA has the cardinality 1:1 and COUNTER has the cardinality 1:N. The third column shows the attributes of the entities.

Generated BRFplus Structures

Table 41

<table>
<thead>
<tr>
<th>Entities</th>
<th>Attributes of the Generated BRFplus Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>CARR</td>
<td>CARR</td>
</tr>
<tr>
<td></td>
<td>CURRCODE</td>
</tr>
<tr>
<td></td>
<td>URL</td>
</tr>
<tr>
<td>CARR_DELTA</td>
<td>URL</td>
</tr>
<tr>
<td>COUNTER</td>
<td>CARR</td>
</tr>
<tr>
<td></td>
<td>COUNT_NUM</td>
</tr>
<tr>
<td></td>
<td>AIRPORT</td>
</tr>
<tr>
<td></td>
<td>C_URL</td>
</tr>
</tbody>
</table>

In the data model, the entities with storage and use type 4 have their own attributes. In contrast, in generated BRFplus structures, the system adds the attributes of the entities with storage and use type 4 with cardinality 1:1 to the attributes of entity type 1. For this reason, the CARR structure has the additional URL attribute.

- Derivations for entities with cardinality 1:1
  - UIs based on Web Dynpro application USMD_ENTITY_VALUE2
If you want to create a derivation for CARR_DELTA in BRFplus, you can execute this only with the function DERIVE_CARR, but not with the DERIVE_CARR_DELTA function. When entities have the cardinality 1:1, in BRFplus you access derivations using the entity type 1.

- UIs based on the Convenience API CL_USMD_CONV_SOM_GOV_API

If you want to create a derivation for CARR_DELTA in BRFplus, you can execute this only with the function DERIVE_CARR_DELTA, but not with the DERIVE_CARR function.

- The generated BRFplus structure of the entity type 1 contains the attributes of the entity type 4 with cardinality 1:1. If you specify the signature for the BRFplus function, you must use the structure of the type-1 entity CARR.

- In contrast, to access an entity type 4 with cardinality 1:N (for example, COUNTER), you call the DERIVE_COUNTER function.

- The signature of the BRFplus function DERIVE_COUNTER is the COUNTER structure.

5.3.2.1 Derivation of the Reconciliation Account After Selecting a Specific Company Code

This document describes how to derivate the value of a field if you enter a specific value into another field: In the example entering the company code 1000 leads to the derivation of reconciliation account with the value 31000.

To access an entity type 4 that has a cardinality 1:1 (for example BP_CENTRL), you have to define the derivation DERIVE_BP_CENTRL to access elements such as Search Term 1 (BU_SORT1).

To access an entity type 4 that has a cardinality 1:N (for example BP_CUS_CC) you have to define the derivation DERIVE_BP_CUS_CC to access elements such as Reconciliation Account (CUS_AKONT).

Note
For more information about derivation, see Definition of Validations and Derivations [page 82].

Activities

To access the entity BP_CUS_CC you must define the DERIVE_BP_CUS_CC function as described below.

1. In the Customizing for Master Data Governance, run the activity Define Validation and Derivation Rules under Master Data Governance » General Settings » Data Quality and Search » Validations and Enrichments » Define Validation and Derivation Rules.
2. Enter the Data model BP and choose Continue.
3. On the Business Rule Framework plus screen, select the Catalog tab, expand the Trigger Function node, and select Derivation. In the context menu choose Create Object Node » Create Function.
4. In the Create Function dialog box, enter the data shown below, choose Create and Navigate to Object, and confirm the dialog box:
   ○ Name: DERIVE_BP_CUS_CC

Note
The naming convention for the function is DERIVE_<name of the entity>.
5. On the **Signature** tab, choose **Add Existing Data Object**.

6. In the **Object Query** dialog box, enter the search criteria shown below and choose **Search**.
   - **Application**: `FMDM_MODEL_BP`
   - **Data Object Type**: `Any`
   - **Name**: `BP_CUS_CC`

7. In the results list, mark `BP_CUS_CC` and choose **OK**.

8. On the **Assigned Rulesets** tab, choose **Create Ruleset**.

9. In the **Create Ruleset** dialog box, enter the data shown below, choose **Create and Navigate to Object**, and confirm the dialog box:
   - **Name**: `BP_CUS_CC_RULESET`
   - **Short Text**: `BP_CUS_CC_RULESET`
   - **Text**: `BP_CUS_CC_RULESET`
   - **Application**: `FMDM_MODEL_BP`

10. In the **Rules** group box, choose **Insert Rule > Create**.

11. In the **Rule** dialog box, choose **Assign Condition > Use Value Range From > Select Context Parameter**.

12. In the **Context Query** dialog box, expand the `BP_CUS_CC` node, mark `CoCode`, and choose **OK**.

13. Select **is equal to** and enter **1000** using the input help.

   **Note**
   Entering the value without the input help may cause inconsistencies due to the wrong data format.

14. Select **Then > Add > Assign Value to Context > Context > More**.

15. In the **Context Query** dialog box, expand the `BP_CUS_CC` node, mark `Recon.acct`, and choose **OK**.

16. In the **Rule**: dialog box, enter **31000** using the input help into the **Change value of BP_CUS_CC to** field and choose **OK**.

   **Note**
   Entering the value without the input help may cause inconsistencies due to the wrong data format.

17. Select **Save**, choose **Activate**, and on the **Confirm Activation** dialog box choose **Activate**.

### 5.3.3 Data Quality Remediation Configuration Guide

This document describes the necessary configuration steps you have to process to use the data quality remediation in Master Data Governance.
Prerequisites

1. You have installed a data quality tool in your system landscape.
2. You have installed a data quality connector in the MDG system. The data quality connector consists of an ABAP class that implements the interface IF_MDG_DQR_DQ_SERVICE and, optionally, an implementation of the BAdl Event Handling for Data Quality Remediation. For more information, see the documentation of this BAdl under General Settings Data Quality and Search Business Add-Ins.

Process

Configuration Process

1. You need to execute the Customizing activity Define Data Quality Service. For more information, see the documentation of this customizing activity under General Settings Data Quality and Search Data Quality Remediation.
2. Copy the Floor Plan Manager (FPM) Configuration Templates
   1. In the package MDG_DQR_GENERAL select Web Dynpro Application configuration MDG_DQR_OVP and choose Start Configurator.
   2. In the Editor for the Web Dynpro ABAP Application Configuration select Continue in Display Mode.
   3. In the Application Configuration MDG_DQR_OVP select the configuration name MDG_DQR_OVP of the component FPM_OVP_COMPONENT.
   4. In the Component Configuration of component MDG_DQR_OVP select Additional Functions Deep Copy.
   5. Choose Configurable Components to expand the list of configurations. To create a copy of the Application Configuration MDG_DQR_OVP, the Overview Page Floorplan MDG_DQR_OVP and at least of the component configuration MDG_DQR_FAILED_REC_LIST_UIBB, set the according flags in the column Copy and enter appropriate names in the column Target Configuration. Then choose Deep-Copy.

   Note
   The technical names of target configurations should reflect the business object type and the data quality service.

   6. Choose a package for the copied configurations.
3. Adapt the Copied Configurations of the FPM
   1. Select the name of the copied application configuration.
   2. Select Edit and enter the ID of the data quality service defined in step 1 in the parameter MDG_DQR_DQS_ID and save.
   3. Select the copied configuration of the component MDG_DQR_OVP.
   4. Change the copy of the configuration MDG_DQR_FAILED_REC_LIST_UIBB by the selecting the respective UIBB and choosing Configure UIBB.
   5. Choose General Settings and then Feeder Class Parameters and enter the Business Object Type for which you want to configure data quality remediation and save your entries.
   6. Configure the List UIBB according to the selected business object type.
4. Configure Change Request Process for DQR
Create a business activity for the business object type with logical action CHANGE_DQR in customizing activity Create Business Activity and assign it to a change request type in Customizing activity Create Change Request Type.

5. To provide the DQR function to your users add the Web Dynpro application MDG_DQR_OVP with the copy of the configuration MDG_DQR_OVP to the role menu.

With the description above you configure the DQR application to include the UI of the data quality tool. The URL to this UI is defined in the Customizing activity Define Data Quality Service. To filter the displayed objects with errors the UI of the data quality tool can raise events that are provided to the DQR application with the implementation of the BAdI Event Handling for Data Quality Remediation. This filtering can also be done with dropdown lists. The configuration template for this variant is provided in the application configuration MDG_DQR_OVP_FILTER and component configuration MDG_DQR_FILTER_UIBB. To configure this variant go through this configuration process and use application configuration MDG_DQR_OVP_FILTER instead of MDG_DQR_OVP in step 2. In this case you need to copy and adapt the component configuration MDG_DQR_FILTER_UIBB by setting the feeder class parameter to the data quality service ID you want to use for your DQR process.

5.4 Process Modeling

The configuration of governance scope, change requests, and workflow offers you flexible ways to model the desired governance process.

Master Data Governance for Customer provides predefined change request types that can be used as examples. For further information, see Configure Change Request Settings [page 18].

5.4.1 Defining a Governance Scope

You can determine a governance scope based on your business needs. Ungoverned fields are read-only in change requests, unless you remove them from the user interface.

**Example**

In the business partner application, you can for example, remove sales grouping data from the governance scope.

**Prerequisites**

You have identified the data models whose governance scope you want to change, as well as the content within each data model that you want to govern.

You are aware of the consequences of changing the governance scope. See the help document in Customizing for Master Data Governance under General Settings > Process Modeling > Define Governance Scope.
Procedure

1. In Customizing for Master Data Governance under | General Settings | Process Modeling | choose Define Governance Scope.

2. In the Data Models view, select the data model whose governance scope you want to define.

3. Make necessary changes to the Governed settings of entity types, attributes, and referencing relationships.
   If there are dependencies, a pop-up informs you of these dependencies and proposes required changes. You can apply required changes or cancel. The following changes to governance scope are not possible:
   - Changes to the Governed setting for entity types with a storage and use type of 1.
     These entity types are shown in the Customizing activity, to enable navigation to attributes.
   - Changes the Governed setting of attributes that are key fields.
     These attributes are not shown in the Customizing activity.
   - Changes to attributes for which the Required Field setting is set to Yes in the data model.
     These attributes are not shown in the Customizing activity.

Result

You have defined a governance scope for the data model. You can keep ungoverned data model elements on the user interface for information purposes. If the elements are not informative to your users, we recommend that you remove them. For more information, see Managing of UI Configurations [page 68].

5.4.2 Configuration of the Change Request Process

When configuring the change request process you need to define the following:

- Processing steps and their processors
- Possible actions of processors
- Process flow between steps
- Change request status in each step

You must configure the following elements:

- Change Request Type
  The change request type defines which data can be processed. The change request type is assigned to one MDG data model and lists the possible entity types that the change request can contain.
  SAP Business Workflow is used to process change requests in SAP Master Data Governance. To define the process flow of the change request you can use standard workflow templates or custom workflow templates when defining a change request type. For more information on SAP Business Workflow, see the Customizing activities under | SAP NetWeaver | Application Server | Business Management | SAP Business Workflow | Alternatively, you can use the MDG rule-based workflow template when defining a change request type. In this case, the content of Business Rule Framework plus (BRFplus) [external document] decision tables defines the process flow of the change request.
  For more information, see the Customizing activity Create Change Request Type under | General Settings | Process Modeling | Change Requests |
• **Change Request Step**
Each change request process consists of a number of change request steps that can be either dialog steps or background steps. For each dialog change request step, you can do the following:
  ○ Assign processors
  ○ Configure validations and data enrichments
  ○ Assign UIs
The processing sequence of the steps is based on the processors’ decision and other criteria that are evaluated by the workflow assigned to the change request type.

If you are not using the rule-based workflow, the workflow template defines the available change request steps. Every change request type using this workflow template can only have the available steps. For more information, see Customizing activity Define Change Request Step Types and Assign Actions under [General Settings > Process Modeling > Workflow](#).

If you are using the rule-based workflow, the Customizing settings and the content of the BRFplus decision tables define the available steps. Every change request type using the rule-based workflow can have different change request steps although all change request types are using the same rule-based workflow template. For more information, see Customizing activity Define Change Request Steps for Rule-Based Workflow under [General Settings > Process Modeling > Workflow > Rule-Based Workflow](#).

• **Change Request Step Type and Change Request Action**
The change request step type defines the possible actions that a processor of a change request step can use. We deliver a number of change request step types, for example Approve Change Request with the possible actions Approve and Reject.

The change request step type of each change request step is determined at runtime. You can configure a change request step that allows the actions Approve and Reject in one case, while allowing Finalize Processing and Send for Revision in another case.

For more information, see Customizing activity Define Change Request Step Types and Assign Actions under [General Settings > Process Modeling > Workflow](#).

• **Change Request Status**
The change request status informs the user about the processing status and determines the possible changes to the change request and the contained data. We deliver a set of status control attributes:
  ○ no processing
  ○ objects can be added or removed
  ○ data changes are allowed
The following statuses that finalize the change request and stop further processing: Final Check Approved and Final Check Rejected. In all other statuses, including any custom statuses, the change request is still open and interlocks the contained data to protect it from processing with other change requests. For more information, see Customizing activity Edit Statuses of Change Requests under [General Settings > Process Modeling > Change Requests](#).

The change request status is set by the workflow. Either the task Set Status of Change Request is used to set the status or, if the rule-based workflow is used, the decision tables are used. For more information, see Customizing activity Configure Workflow Tasks under [General Settings > Process Modeling > Workflow](#).
5.4.2.1 Designing the Change Request Process

For the design of the change request process and its configuration, it is useful to create a diagram that comprises all change request steps and their connections. The recommended process is as follows:

Process

1. Start with step 00 and an appropriate description, for example Request. Provide a name for the group of users that are allowed to create change requests of this type, for example Requester.

   ![Figure 4: Change Request Step: 00/Request. Change Request Type: Requester](image)

   Note
   You control which users can create change requests of a certain type with the authorization object USMD_CREQ. For further information on authorizations, see Authorization Objects and Roles Used by Master Data Governance [external document].

   Also, add a step 99 to represent the end of the process.

   ![Figure 5: Change Request Step: 99/Complete](image)

2. Add a step for each task that a user needs to perform. Assign a step number that is unique for the process and choose an appropriate description. Name the group of users that shall perform the task. Select a step type in Customizing activity Define Change Request Step Types and Assign Actions under General Settings > Process Modeling > Workflow that fits to the task and includes the actions the processor should be able to choose. Add the step type and the possible actions as outcomes to the diagram like shown below.

   ![Figure 6: Dialog Step 90/Approve: With Expert as Processor, Approve Change Request as Step Type, and Approve and Reject as Possible Actions](image)
3. Add a step for each background task. Assign a step number and a description. Add this information together with the description of the background task to the diagram. Also, include all possible outcomes of the task on which you want to react in the process. Some important standard tasks of MDG to work with the change request are the following:

- **ACTIVATE CHANGE REQUEST (TS60808002)**
- **DISCARD CHANGE REQUEST (TS75707936)**
- **CHANGE REQUEST REPLICATION (TS60807976)**

![Figure 7: Background Step 91/Activate: To Activate All Data of the Change Request with Task Activate Change Request and Two Outcomes to Handle Successful and Unsuccessful Completion of the Task](image)

4. Connect each step with an arrow that originates from the respective outcome of the previous step and ends at the step that should follow. For each arrow, add the new status that the change request shall have, when the process proceeds from one step to the next.

![Figure 8: If the expert chooses to approve the change request, the status shall be set to 02/Changes to be Executed, and the system shall activate the change request.](image)

**More Information**

For more information, see Creating a Basic Change Request Process [page 103] and Add User-Agent Steps [page 106] for examples to configure the rule-based workflow.
5.4.2.2  Configuration of the Workflow

SAP Business Workflow is used to process change requests in Master Data Governance (MDG). You have the option to use standard or custom workflow templates when defining a change request type. If you choose standard templates you can customize predefined change request process flows. If you choose custom templates you can create your own process with the workflow builder of SAP Business Workflow.

Alternatively, you can use the MDG rule-based workflow, which is based on one generic workflow template. You can configure your particular change request process with BRFplus decision tables. Using the rule-based workflow you can add or remove a change request step or change the order of the steps without the need to change anything in the workflow template by adapting the BRFplus decision tables.

Prerequisites

You have performed the basic workflow setup as described in the document Workflow Set-Up.

Activities

**Standard Workflow Template**

1. Choose an appropriate template by examining its documentation.
2. Create the change request type and enter the chosen workflow template.
3. Perform further configuration according to the requirements of the template, for example, assign processors to the change request steps.

**Custom Workflow Template**

1. Create the workflow template.
2. Define the change request steps in the MDG Customizing.
3. Create the change request type and enter your custom workflow template.
4. Perform further configuration, according to the requirements of the template, for example assign processors to the change request steps.

**Rule-based workflow**

1. Create the change request type.
2. Define change request steps in MDG Customizing.
3. Create decision tables.

5.4.2.2.1  Workflow Set-Up

You use this process to define the mandatory Customizing settings that are needed to enable SAP Business Workflow for the change request process in Master Data Governance.
Prerequisites

You have defined the necessary settings for SAP Business Workflow and defined the organizational plan in Customizing under SAP NetWeaver > Application Server > Business Management > SAP Business Workflow.

Process

1. The workflow system user (typically WF-BATCH) processes background tasks of MDG. Therefore, this user needs to have the required MDG authorizations. Assign the PFCG role SAP_MDG_WF_ADM to the workflow system user in transaction SU01. For more information, see SAP Note 1650993.

2. Create event type linkages for the business object BUS2250 (MDG Change Request) as described in Customizing activity Activate Type Linkage under General Settings > Process Modeling > Workflow.

3. To assign processors to change request types and change request steps, decide on the possible agents of the MDG workflow tasks in general. In Customizing activity Configure Workflow Tasks under General Settings > Process Modeling > Workflow assign specific agents from your organizational plan to each dialog task. In the attributes pop-up of each dialog task, select to whom processors may forward a respective work item. Instead of assigning specific possible agents to a dialog task, you can also classify a dialog task as general task, so that a work item can be executed by any user. All users in the list of possible agents that are also assigned as processors of a change request step, are selected as the agents at runtime and will receive the work item. Make the settings for all dialog tasks of the application component CA-MDG-AF and the respective components of the MDG application that you use.

   Note
   If you assign a processor to a change request step that is not assigned as possible agent, the workflow will end in an error at runtime unless you have classified the task as general task.

5.4.2.2.2 Rule-Based Workflow

Instead of building your own workflow template, you can use the MDG rule-based workflow. Using the rule-based workflow, you can configure any kind of change request process without having to create and adapt a workflow template. You can define different change request processes in decision tables of the Business Rules Framework plus (BRFplus), which are maintained in Customizing for each change request type. At runtime, the current step, the user interactions, and other parameters in the decision tables determine the process flow of the change request. When you adapt the decision tables in BRFplus, you can add or remove a change request step, or change the order of the steps without changes in the workflow template.

The rule-based workflow uses BRFplus to determine the change request status, the next change request step, and expected agent(s). To make this information available, the system uses the current step, the last action, the priority of the change request, and, where appropriate, the reason of rejection as input parameters. You access the BRFplus application to determine how change requests are processed for a particular change request type in Customizing activity Configure Rule-Based Workflow under General Settings > Process Modeling > Workflow. If you process this Customizing activity for a change request type for the first time, the system generates a BRFplus application for each change request type. Each application contains functions, rulesets, and decision tables. The content of the decision tables defines the change request process.

Three decision tables are available for each change request type:
**Single Value Decision Table**

The *Single Value Decision Table* `DT_SINGLE_VAL_<change request type>` defines the process flow between the change request steps. Based on the previous step, the action, and other parameters, this table returns the next step and other result parameters. The most important result parameter is the condition alias that links to the other decision tables. This decision table has the following condition columns:

- **CR Previous Step**
  This parameter contains the previously processed change request step.

- **Previous Action**
  This parameter contains the result of the previous system or previous user action.

- **Chng. Req. Priority**
  This parameter contains the current priority of the change request.

- **Chng. Req. Reason**
  This parameter contains the reason for this change request.

- **CR Rejection Reason**
  This parameter contains the reason for rejection of this change request.

- **CR Parent Step** and **Parallel Agt Grp No.**
  These columns are used for parallel processing. Based on the columns, the rule-based workflow determines the next step in the relevant subprocess. The system identifies the relevant subprocess by referring to the values in **CR Parent Step** and **Parallel Agt Grp No.**. For more information, see Parallel Processing [page 108].

Based on the data from these condition columns, the system carries out the actions and sets the statuses outlined in the result columns. This decision table has the following result columns:

- **Condition Alias**
  The condition alias references the other decision tables. Each condition alias must be handled using at least one row in either the *User Agent Decision Table* or the *Non-User Agent Decision Table*.

- **New Chng. Req. Step**
  This parameter contains the next step in the process.

- **New CR Status**
  This parameter contains the new status for the change request.

- **Hours to Completion**
  Once the expected completion time in hours has passed without any of the agents having processed the work item, the system automatically sends a notification.

- **Merge Type** and **Merge Parameter**
  In parallel processing, the merge type and the merge parameter define how the results of the subprocesses are merged back into the higher-level process. The system only supports the merge type calling a BAdI method. The filter value for the BAdI is determined by the merge parameter. For more information, see Parallel Processing [page 108].

- **Dyn Agt Sel Service**
  The service name is used to select an implementation of BAdI: Dynamic Selection of Agent in Rule-Based Workflow in MDG Customizing under *General Settings > Process Modeling > Workflow > Rule-Based Workflow > Business Add-Ins*. The implementation can overwrite various result values and determine the user agent groups. You can use this BadI, for example, to determine the processors at runtime based on data in the change request. For more information, see the documentation of BAdI: Dynamic Selection of Agent in Rule-Based Workflow.
### User Agent Decision Table

The **User Agent Decision Table** `DT_USER_AGT_GRP_<change request type>` determines the processors of the change request step that was returned as the next step by the **Single Value Decision Table**. It also determines the change request step type that defines the actions processors can execute.

This table has the following condition column:

- **Condition Alias**
  
  The condition alias links the row in this table with corresponding rows in the **Single Value Decision Table**.

This decision table has the following result columns:

- **User Agt Grp No.**
  
  Enter an arbitrary value in the column **User Agt Grp No.**, and enter the processors in the column **User Agent Value**. If you need more than one entry for **User Agent Value** to define the processors, enter the same value for **User Agent Group** and **Condition Alias** in each row to create one user agent group.
  
  You configure parallel processing of the change request step by entering different values for **User Agent Grp No.** and the same condition alias. For each value in **User Agent Grp No.**, a separate subprocess is started. For more information, see **Parallel Processing** [page 108].

- **Step Type**
  
  The step type defines the possible actions for the processor in the change request step.

- **User Agent Type** and **User Agent Value**
  
  Identifies what kind of agent receives the work item in a change request. It can be a single user, an organizational unit, a role, a job, a position, or a special user. The user agent value defines the agents a work item can be sent to. It can be a user ID, or a user group ID. It can also point to rather than directly identify the user agent – for example, with user agent type `SU` (single user), the user agent value `LAST` specifies the last processor, and the user agent value `INIT` specifies the requester of the change request.

### Non-User Agent Decision Table

This decision table determines the process patterns for background steps.

The **Non-User Agent Decision Table** `DT_NON_USER_AGT_GRP_<change request type>` contains the background steps that are involved in the change request process and that do not have end user participation.

This table has the following condition column:

- **Condition Alias**
  
  The condition alias links the row in this table with corresponding rows in the **Single Value Decision Table**.

This decision table has the following result columns:

- **Agent Group**
  
  Enter an arbitrary value in this column to execute the operation in column **Process Pattern** in the background.
  
  If you are using parallel processing, create a row for each process pattern that is to be executed in a separate subprocess. Choose a different value in this column for each row. For more information, see **Parallel Processing** [page 108].

- **Process Pattern**
  
  The **Process Pattern** controls the flow of the process and to define what the system is to perform in this change request step. Frequently used values are:

  - **05 Activation (do not bypass snapshot)**
    
    Activates the change request, for example, after final approval.
08 **Roll Back Change Request**
Removes all inactive data, for example, after the change request was rejected.

99 **Complete (Sub-)Workflow**
Completes a workflow or a sub-workflow. This process pattern is used, for example, in the last step to end the change request process.

See [Process Pattern](#page102) for a complete list of available process patterns.

**Service Name**
The meaning of this parameter depends on the process pattern. For example, it contains the workflow template when creating a sub-workflow with process pattern 03 **Call Sub-Workflow**.

More Information

For information on how to create and enhance your rule-based workflow, see [Creating a Basic Change Request Process](#page103) and [Add User-Agent Steps](#page106).

### 5.4.2.2.2.1 Configuring the Rule-Based Workflow

This document explains how to configure the rule-based workflow for a change request process that you have described using a process diagram as explained in [Designing the Change Request Process](#page92).

**Prerequisites**

- You have completed the Customizing settings as described in [Workflow Set-Up](#page94).
- You have created a diagram of the change request process that you want to configure as described in [Designing the Change Request Process](#page92).

**Process**

- **Enhance Process Diagram**
  Enhance the process diagram with further information required by the rule-based workflow. For each non-user agent change request step, determine the appropriate [Process Pattern](#page102) and add the information to the diagram.
To activate the change request, you need to use the process pattern 06/Activation. For each arrow pointing to a change request step, choose a 3 digit identifier for the condition alias. It is common to use abbreviations of the step’s meaning for better readability, for example `APP` for an approval step.

For information about an example of a process diagram that is enhanced for the rule-based workflow, see Add User-Agent Steps [page 106].

- **Create Change Request Type**
  In Customizing activity `Create Change Request Type` under `General Settings` > `Process Modeling` > `Change Requests`, create the change request type for which you want to define the process flow. Assign the rule-based workflow template `WS60800086` to the change request type.

- **Define Change Request Steps**
In Customizing activity Define Change Request Steps for Rule-Based Workflow under General Settings Process Modeling Workflow Rule-Based Workflow define the process steps that are used in the process diagram of your change request type.

- **Service Names**
  In the case of a complex workflow scenario, for example, when using a handler to merge the results of parallel processing, you need to define service names for the BAdI implementations that you need to use. For more information, see the documentation of Customizing activity Define Service Names for Rule-Based Workflow under General Settings Process Modeling Workflow Rule-Based Workflow.

- **Build Decision Tables**
  Start Customizing activity Configure Rule-Based Workflow under General Settings Process Modeling Workflow Rule-Based Workflow and enter your change request type to open the BRFplus workbench and to enter the values for the decision tables.

  **Note**
  If you perform this activity the first time for this change request type, the BRFplus application is generated. Depending on the settings of the client, you are asked to assign a transport request and a software component.

1. For each arrow in your process diagram, enter a row in the Single Value Decision Table DT_SINGLE_VAL_<change request type>. Use the step numbers on each end of the arrow as the values for **CR Previous Step** and **New Chng. Req. Step**. The action code of the previous step that triggers this connection is the value for **Previous Action**. The labels on the arrow provide the values for **Condition Alias** and **New CR Status**.

   ![Diagram](https://example.com/diagram.png)

   Figure 11: The arrow of this diagram leads to the following values in the decision table: CR Previous Step = 90. New Chng. Req. Step. = 91. Previous Action = 03. Condition Alias = ACT. New CR Status = 02.

   **Note**
   You can use the condition columns **Chng. Req. Priority**, **Chng. Req. Reason**, **CR Rejection Reason**, **CR Parent Step**, and **Parallel Agt Grp No.** as additional parameters to make the process flow more specific.

   You can enter a time limit for the processors of the next change request step in **Hours to Completion**. This uses the feature of the requested end deadline monitoring of the SAP Business Workflow. The rule-based workflow will send a notification to all processors of this change request step as a reminder to complete this task.

   The result columns **Merge Type** and **Merge Parameter** are used for parallel processing. For further information, see Parallel Processing [page 108].
Instead of providing values for the result columns in the decision table, you can provide a service name in Dyn Agt Sel Service to link to an implementation of BAdI: Dynamic Selection of Agent in Rule-Based Workflow. For more information, see the documentation of BAdI: Dynamic Selection of Agent in Rule-Based Workflow in MDG Customizing under General Settings > Process Modeling > Workflow > Rule-Based Workflow > Business Add-Ins.

2. For each user agent step in your process diagram, enter a row in the User Agent Decision Table DT_USER_AGT_GRP_<change request type>. If you followed the recommendation in Designing the Change Request Process [page 92] to use the same condition alias for all arrows that point to a change request step, use this value for the column Condition Alias. If you use different aliases, you need to create multiple rows, one for each alias.

Transfer the values for Step Type, User Agent Type, and User Agent Value from the diagram into the table. The valid values for User Agent Type and User Agent Value are defined by your organizational structure (for example, see Customizing activity Edit Organizational Plan) and identify an organizational object, for example, the purchasing department. If you use SU for User Agent Type you can use INIT (Initiator) as User Agent Value to select the requester of the change request as processor. Furthermore, the value LAST for User Agent Value selects the processor of the previous step as the processor.

If the overall group of processors for the change request step consists of multiple organizational objects, create a row for each object. In this case and unless you want to configure parallel processing of the change request step, use the same value for User Agt Grp No. for this condition alias.

You configure parallel processing of the change request step by using different values for User Agt Grp No. for the same condition alias. For further information, see Parallel Processing [page 108].

3. For each background step in your process diagram, enter a row in the Non-User Agent Decision Table DT_NON_USER_AGT_GRP_<change request type>.

If you followed the recommendation in Designing the Change Request Process [page 92] to use the same condition alias for all arrows that point to a change request step, use this value for the column Condition Alias. If you use different aliases, you need to create multiple rows, one for each alias.

Transfer the value for Process Pattern from the diagram into the table. If required by the chosen process pattern, specify the Service Name. Unless you want to configure parallel processing in this change request step, choose any value for Agent Group, for example 001. For more information, see Parallel Processing [page 108].

Figure 12: The information from this diagram leads to the following values in the decision table: Condition Alias = APP. User Agt Grp No. = 001 (arbitrary value). Step Type = 02. User Agent Type = AG. User Agent Value = MD Experts (assuming there is a PFCG role named MD Experts and all users assigned to this role should be processors).
Figure 13: The information from this diagram leads to the following values in the decision table: Condition Alias = ACT. Process Pattern = 06. Service Name = <not required>. Agent Group = 001 (arbitrary value).

Caution
The decision tables are processed in sequence Therefore, the table entries should be arranged starting with the most specific ones, followed by more general ones.

More Information
For information about examples of process diagrams related to the rule-based workflow, see Creating a Basic Change Request Process [page 103] and Add User-Agent Steps [page 106].

5.4.2.2.2.2 Process Pattern

The rule-based workflow groups several workflow steps together to form basic operations that are called Process Patterns. These patterns are used to control the flow of the change request process or to define which background task the system will perform in a change request step.

Technically, the rule-based workflow runs in a loop. In each repetition of the loop, one out of several process patterns is executed. The workflow continues to run in this loop until the change request process is ended with the process pattern 99 Complete (Sub-)Workflow.

If the current change request step is a user-agent step, the used process pattern is 01 UI Dialog. For non-user agent steps, the column Process Pattern in the Non-User Agent Decision Table DT_NON_USER_AGT_GRP_<change request type> is used to determine the pattern.

The possible process patterns are:

- 01 UI Dialog
  
  This process pattern is used by the system for user-agent change request steps and should not be entered by you in the Non-User Agent Decision Table. It is a special process pattern that is always automatically selected if a user agent has been found in the user agent decision table. This process pattern uses the dialog task Dialog Processing TS60807954.

- 02 Call Synchronous Method

PUBLIC © Copyright 2016 SAP SE or an SAP affiliate company. All rights reserved.
You can use this process pattern to include operations that are not provided from SAP. This process pattern uses the background task Synch. System Method TS60807949. For more information, see BAdI: Calling of System Method for Rule-Based Workflow in MDG Customizing under General Settings > Process Modeling > Workflow > Rule-Based Workflow > Business Add-Ins.

- **03 Call Sub-Workflow**
  You can use this process pattern to start a sub-workflow. The background task Subworkflow for Single Step Workflow TS60807994 starts a sub-workflow with the workflow template ID that is read from the column Service Name of the non-user agent decision table.

- **04 Call Data Replication**
  You can use this process pattern to start the replication of the master data after the change request has been activated. This process pattern uses the background task Change Request Replication TS60807976 and the method DISTRIBUTE of the object type MDG Change Request BUS2250 to replicate the object using the data replication framework (DRF).

- **05 Activation (do not bypass snapshot)**
  You can use this process pattern to activate the data in the change request. This process pattern uses the background task Activate Change Request TS60808002 with the indicator IGNORE_SNAPSHOT_DIFF not set. The value of Previous Action is updated with the result of the operation enabling you to handle error situations. If there have been conflicting changes to the data in the standard master data tables while the change request was in process the activation fails. In this case, Previous Action is set to 33 Activation failed for Snapshot. If the activation was successful Previous Action is set to 31 Activation Successful. In all other cases, Previous Action is set to 32 Activation failed.

- **06 Activation (bypass snapshot)**
  You can use this process pattern to activate the data in the change request, even if the data has been changed in the backend since the change request was created. The system ignores these potential changes and overwrites them. This process pattern uses the background task Activate Change Request TS60808002 with the indicator IGNORE_SNAPSHOT_DIFF set.

- **07 Validate Change Request**
  You can use this process pattern to validate the change request data. The results are written to the application log. The process pattern uses the background task Check Change Request TS75707952.

- **08 Roll Back Change Request**
  You can use this process pattern to remove the inactive data of the change request from the staging area if the change request should not be activated. This process pattern also provides the information when and by whom the change request was released and sets the change request status to 06 Final Check Rejected. The process pattern uses the background task Discard Change Request TS75707936.

- **09 Error**
  You can use this process pattern to handle errors and exceptions. The process pattern uses the background task Error Handler TS60807951.

- **99 Complete (Sub-)Workflow**
  You can use this process pattern to end the rule-based workflow instead of looping back.

### 5.4.2.2.2.3 Creating a Basic Change Request Process

This document describes how to enable a basic change request process using the MDG rule-based workflow. This basic change request process only activates the change request after it was submitted. The process does not
include any dialog step. To provide data governance capabilities, you need to enhance the process adding further change request steps such as approving the change request.

The figure in this document shows a complete process.

The process starts with step 00 when the requester submits the change request. The next step 91 is the activation of the change request. If the change request is successfully activated, its status is set to Final Check Approved and the process ends with step 99. If the activation fails, the change request is rolled back in step 92, the change request status is set to Final Check Rejected, and the process ends.

**Prerequisites**

You have created a change request type and you have entered the template for rule-based workflow WS60800086 in Customizing activity Create Change Request Type under General Settings > Process Modeling > Change Requests. In the following example configuration, the change request type CR_TYPE is used.

**Process**

You need to perform the following steps in order to configure the rule-based workflow for the basic change request process:

1. Create necessary change request steps.
   
   Define the change request steps 00, 91, 92, and 99 as shown in the figure in Customizing activity Define Change Request Steps for Rule-Based Workflow under General Settings > Process Modeling > Workflow > Rule-Based Workflow.
2. Define decision tables.

For every change request type, there is a separate set of BRFplus decision tables that contain the configuration of the change request process. You can start the configuration of the rule-based workflow in Customizing activity Configure Rule-Based Workflow under General Settings → Process Modeling → Workflow → Rule-Based Workflow.

In the Single Value table of change request type CR_TYPE, you define the sequence of the steps. If a column is not mentioned in the tables below it is not relevant for this process configuration. You have to add a row in the Single Value table for each arrow in the figure to connect two change request steps and use the following information from the figure:

- Previous Change Request Step
- New Change Request Step
- Change Request Status
- Change Request Action
- Condition Alias

The first row of the Single Value table corresponds to the arrow from step 00 to step 91 in the figure: The column CR Previous Step contains 00 and the column New CR Step contains 91. Since the first change request step is the request step that produces no action result, the column CR Previous Action is left empty. The column New CR Status contains 02 as the status of the change request in step 91. Finally, the column Condition Alias contains the identifier ACT that you need to assign and that is used to connect this row with rows in the other decision tables.

**Single Value table**

<table>
<thead>
<tr>
<th>CR Previous Step</th>
<th>CR Previous Action</th>
<th>Condition Alias</th>
<th>New CR Step</th>
<th>New CR Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>n.a.</td>
<td>ACT</td>
<td>91</td>
<td>02</td>
</tr>
<tr>
<td>91</td>
<td>31</td>
<td>END</td>
<td>99</td>
<td>05</td>
</tr>
<tr>
<td>91</td>
<td>31</td>
<td>RB</td>
<td>92</td>
<td>02</td>
</tr>
<tr>
<td>92</td>
<td>n.a.</td>
<td>END</td>
<td>99</td>
<td>06</td>
</tr>
</tbody>
</table>

The basic process contains only steps with background steps. Therefore, you only have to configure the Non User Agent table and the User Agent table is left empty. In the figure all arrows pointing to the same change request step have identical condition aliases. These condition aliases have been chosen to match the process pattern of this step.

You have to add a row in the Non User Agent table for each change request step and use the following information from the figure:

- Condition Alias
- Process Pattern
The column Agent Group is only relevant for parallel processing. Use the value 001 to create one work item for a change request step.

If you look at the arrow with condition alias ACT from step 00 to step 91 with process pattern 05, the first row in the Non User Agent table contains the condition alias ACT, agent group 001 and process pattern 05. The following rows are needed in the Non User Agent table for the configuration of the complete basic process:

<table>
<thead>
<tr>
<th>Condition Alias</th>
<th>Agent Group</th>
<th>Process Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACT</td>
<td>001</td>
<td>05</td>
</tr>
<tr>
<td>RB</td>
<td>001</td>
<td>08</td>
</tr>
<tr>
<td>END</td>
<td>001</td>
<td>99</td>
</tr>
</tbody>
</table>

After you have saved and activated the new entries for the Single Value table and the Non User Agent table, you can use the new change request type.

### 5.4.2.2.2.4 Add User-Agent Steps

This document describes how to enhance the basic change request process with a user agent step. In the basic process, a change request is immediately activated after the requester submits the change request without further involvement of another user. In this enhanced process, a second user checks the change request in an additional user-agent step. If this user decides to approve the change request, the activation is started with change request step 91. Otherwise, the roll back of the change request is started with change request step 92. The other change request steps are not changed.

The terms dialog step and user agent step are used as synonyms in MDG.

To enhance the basic process from the document Creating a Basic Change Request Process [page 103] to the enhanced process described in this document, the new step 90 Final Check with step type 2 Approve Change Request is added. The user symbol next to the step type indicates that this is a user-agent step. The arrow from change request step 00 now points to the new change request step 90. The condition alias of this arrow was chosen as FC to abbreviate Final Check. The arrow, depicting that the user has accepted the change request with action 03, points to the change request step 91. The condition alias ACT for the change request step 91 is added to the arrow. The arrow, depicting that the user has rejected the change request with action 04, points to the change request step 92. The condition alias RB for the change request step 92 is added to the arrow.
Prerequisites

You have configured the rule-based workflow for the basic change request process, as described in Creating a Basic Change Request Process [page 103]. In the following example process, the change request type CR_TYPE and the user FINAL_CHECK_USER are used.

Process

You need to process the following steps in order to extend the basic workflow with a user step:

1. Create the new change request step.

   The new change request step for the user dialog is defined in Customizing activity Define Change Request Steps for Ruled-Based Workflow under General Settings > Process Modeling > Workflow > Ruled-Based Workflow.

   Workflow Step Numbers

   Table 45

<table>
<thead>
<tr>
<th>Type of Chg. Request</th>
<th>CR Step</th>
<th>Keys</th>
<th>Validation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR_TYPE</td>
<td>90</td>
<td>n.a.</td>
<td>n.a.</td>
<td>Final Check</td>
</tr>
</tbody>
</table>

2. Adapt and add lines to decision tables.

   Comparing the figure Change Request Process Including Dialog Tasks with the figure Basic MDG Change Request Process of the basic rule-based workflow you can see that you have to add new rows to the decision table and also change existing rows of the decision table, because the first arrow from change request step 00 to step 91 in figure Basic MDG Change Request Process has changed. In the figure Change Request Process Including Dialog Tasks, the arrow points to the new change request step 90. The Single Value table row with the previous change request step 00 has changed to the following:

Figure 15: Change Request Process Including a User Agent Step
Table 46

<table>
<thead>
<tr>
<th>CR Previous Step</th>
<th>CR Previous Action</th>
<th>Condition Alias</th>
<th>New CR Step</th>
<th>New CR Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>n.a.</td>
<td>FC</td>
<td>90</td>
<td>02</td>
</tr>
</tbody>
</table>

After this change, you have to add a new row to the *Single Value* table for every arrow that is depicted in the figure *Change Request Process Including Dialog Tasks* and not depicted in the figure *Basic MDG Change Request Process*. You have to add the following rows to configure the new sequence of steps:

Table 47

<table>
<thead>
<tr>
<th>CR Previous Step</th>
<th>CR Previous Action</th>
<th>Condition Alias</th>
<th>New CR Step</th>
<th>New CR Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>90</td>
<td>03</td>
<td>ACT</td>
<td>91</td>
<td>02</td>
</tr>
<tr>
<td>90</td>
<td>04</td>
<td>RB</td>
<td>92</td>
<td>02</td>
</tr>
</tbody>
</table>

In the basic rule-based workflow, only background tasks are used. In the enhanced workflow described in this document, a dialog task is used. In the *User Agent* table, you have to configure the user agent group, the change request step type, the user agent and the user agent value for the new change request step 90. The following line with the condition alias FC for the new change request step is required:

Table 48

<table>
<thead>
<tr>
<th>Condition Alias</th>
<th>User Agt Group</th>
<th>Step Type</th>
<th>User Agent Type</th>
<th>User Agent Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FC</td>
<td>001</td>
<td>02</td>
<td>US (User)</td>
<td>FINAL_CHECK_USER</td>
</tr>
</tbody>
</table>

5.4.2.2.2.5 Parallel Processing

The rule-based workflow allows the parallel processing of a change request for processors belonging to more than one agent group. For example, you can define an approval step in which both one processor of the controlling department and one processor of the purchasing department need to approve the change request. Both groups of users will receive a work item for the processing of the change request at the same time and can complete their work independent of each other.

**Parent Step**

The step from which parallel processing starts is called parent step. In contrast to regular change request steps, you assign multiple agent groups to a parent step. For each assigned agent group, a subprocess is started that is executed in parallel. The first step of each subprocess has the same step number as the parent step. Therefore, the step number of the parent step and the agent group number of the subprocess are additionally used to uniquely identify each step in subprocesses. The process that is started initially after the change request was submitted is called root process.

**Process Flow in Sub-Processes**

Each subprocess is an instance of the rule-based workflow. Subprocesses provide the information of their parent step and the agent group for which they were started. This information is used during the evaluation of the single value decision table for determining the next change request step. Using these parameters in the single value decision table, you separate the configuration of the process flow in the initial process from the process flow in the subprocesses.

**Ending of SubProcesses**

Subprocesses have to be ended by using a change request step with the process pattern 99 *Complete (Sub-)Process*. 

PUBLIC
© Copyright 2016 SAP SE or an SAP affiliate company.
All rights reserved.

Master Data Governance for Customer
Adapting Master Data Governance for Customer
When all subprocesses have ended, processing continues in the parent step by evaluating the action results of the subprocesses. This is done by a result handler. For example, if any user of the two departments chooses to reject the change request in a subprocess that the overall result of the parent step rejects the change request. Result handlers are implementations of BAdI: Handling of Parallel Results in Rule-Based Workflow in MDG Customizing under General Settings > Process Modeling > Workflow > Rule-Based Workflow > Business Add-Ins and referenced by their service name.

For more information, see the documentation of BAdI: Handling of Parallel Results in Rule-Based Workflow.

Using the actions and steps returned by the subprocesses, the result handler returns a merge step and merge action that are used in the next loop of the rule-based workflow to evaluate the single value decision table.

You need to specify the result handler in the row of the single value decision table that leads to the parent step of the subprocesses. This is done by providing the value B in column Merge Type and the service name of the result handler in column Merge Parameter.

Agent Groups

Agent groups are assigned to change request steps through the condition alias of the single value decision table. User agent groups are defined in the user agent decision table for dialog steps. Non-user agent groups are defined in the non-user agent decision table for background steps. Both types of groups are uniquely identified by their group number and the condition alias.

User Agent Groups

A user agent group specifies the assigned processors of a change request step and the step type of the dialog step. All users assigned to a user agent group will receive a workitem to process the change request step. You can use multiple organizational objects to specify the members of the user agent group. In this case, you need to create a row for each organizational object in the user agent decision table and use the same value in the columns Condition Alias and User Agent Grp No.. This defines a user agent group with multiple rows.

You configure the parallel processing of a change request step by entering different values for User Agent Grp No. and the same condition alias. For each value in User Agent Grp No., a separate subprocess is started.

It is not allowed to have rows with the same condition alias and the same user agent group, but different step type, because a change request step can only have one change request step type. However, it is possible to configure parallel steps that have different change request step types.

Non-User Agent Groups

A non-user agent group specifies the process pattern that should be executed in the background in a change request step. A non-user agent group is defined by entering the condition alias, the agent group, and the process pattern in the non-user agent group decision table.

You configure parallel processing of a change request step by entering different values for the agent group and the same condition alias. For each value in the agent group, a separate subprocess is started to execute the respective process pattern.

It is not allowed to have more than one row with the same values for the condition alias and the agent group, because only one process pattern can be executed in each change request step. However, you can define parallel background steps, in which process patterns are executed in parallel.

It is not allowed to have a row in the non-user agent decision table that has the same values for condition alias and agent group as a row on the user agent decision table, because a change request step can only be either a dialog or a background step. However, you can define two parallel steps, one as a dialog step and the other as a background step.

Phases of Parallel Processing

The phases in which the rule-based workflow handles parallel processing are as follows:

1. After having evaluated the decision tables, it is checked whether there is more than one agent group assigned to the change request step.
2. For each agent group, a new instance of the rule-based workflow is started. The already determined agent group and the step number of the parent step are passed to the instance. In the parent step, the processing is suspended until every subprocess has ended.

3. In the initial loop of the rule-based workflow, the agent group is already known and processing can directly continue by creating the workitem for the dialog step or executing the process pattern in case of a background step. After that, a new loop is started.

4. In the second loop, the action result of the previous step, the information of the parent step, and the agent group of this sub-process are used to find a matching row in the single value decision table and to find the assigned condition alias in the user agent decision tables and non-user Agent decision tables.

5. If a step with process pattern 99 End (Sub-)Process is found, the workflow ends and control returns to the parent step. If there are further steps defined for the subprocess they are processed in further loops of the subworkflow.

6. After all subprocesses have ended, the result handler is called. It uses the action results’ return and the change request step numbers’ return by the subprocesses to determine a merge step and merge action. Both values are used in the next loop of the rule-based workflow to query the decision tables and processing continues until the root process is ended as well.

More Information

Rule-Based Workflow: Technical Details [external document]

5.4.2.2.2.6 Rule-Based Workflow: Technical Details

This document explains how the rule-based workflow works by describing the workflow template of the rule-based workflow and how this workflow template of the rule-based workflow uses the BRFplus application of a particular change request type.

We deliver the standard workflow template WS60800086 for the rule-based workflow. This workflow template consists of the following steps:

1. **Start Workflow**
   An instance of the rule-based workflow template is started when a user submits a change request of a type that has the rule-based workflow template assigned. The same workflow template is also used to create sub-workflow instances for parallel processing.

2. **Determine Change Request Type**
   The system determines the change request type; for example, Create Business Partner or Change Business Partner and stores the change request type in the workflow container.

3. **Check Assignment of Processor to Workflow**
   The system checks whether a processor is already assigned to the workflow, for example, the current workflow instance is a sub-workflow that was started for parallel processing.
   
   If a processor is not yet assigned, the system launches BRFplus. The BRFplus decision tables for the change request type are used to find the next step, the process pattern, and the agents, based on the previous step and action. If the current workflow instance is the main workflow, the system also refreshes the status of the change request.

4. **Determine Whether Single Processing or Parallel Processing is Configured**
The system determines the number of configured agent groups of the current change request step. An agent group can consist of a single user or multiple users. For example, it might be necessary that users in the purchasing department and users in the accounting department should be able to approve the change request in parallel.

If more than one agent group is found, parallel processing is configured and the system proceeds as follows:

1. The system creates multiple workflow instances of the WS60800086 template: one for each agent group. These sub-workflows run in parallel.
2. As soon as all subworkflows are completed, the BAdI: Handling of Parallel Results in Rule-Based Workflow in MDG Customizing under General Settings Process Modeling Workflow Rule-Based Workflow Business Add-Ins is called in order to merge the results of the parallel subworkflows into one result and, based on those results, determines the next step of the change request process.

5. **Branch by Process Pattern**

   Based on the determined process pattern, the workflow branches into one out of several basic operations of the rule-based workflow.

   For more information, see Process Pattern [page 102]

6. **Check Workflow Completion**

   The system checks whether the process pattern was Complete (Sub-)Workflow.
   - If this is the case the system completes the workflow.
   - If this is not the case the system returns to step 3 and starts again.

### 5.4.3 Scope for Hierarchy-Specific Changes

You can determine the extent to which users can make parallel changes to a hierarchy that belongs to a particular hierarchy type. A change to a hierarchy can comprise adding a node, moving a node, removing a node, changing the attributes of a node, or creating a hierarchy. After a change to a hierarchy is saved to a change request, changes to interlocked nodes must be saved to the same change request. The system determines which nodes are interlocked by referring to the Interlocking setting in Customizing for the relevant hierarchy type.

Hierarchy nodes that represent business objects are technically distinct from the business objects themselves. Interlocking affects the parallel processing of hierarchy nodes only.

#### The Interlocking Setting

You can define the scope of interlocking in Customizing for Master Data Governance under Process Modeling Hierarchies Define Scope for Changes

The Interlocking setting applies to a Hierarchy Type and specifies which nodes besides the node being changed are interlocked while a hierarchy-specific change is in process. The setting is described in the table below:

<table>
<thead>
<tr>
<th>Interlocking Setting</th>
<th>Interlocked Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loose</td>
<td>Nodes assigned to the parent node of the node being changed.</td>
</tr>
<tr>
<td>Strict</td>
<td>Interlocking propagates upwards and downwards from the parent node of the node being changed.</td>
</tr>
</tbody>
</table>
**Interlocking Setting**

<table>
<thead>
<tr>
<th>Interlocking Setting</th>
<th>Interlocked Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>● <strong>Upwards interlocking</strong> interlocks the parent node and its assigned nodes, the parent node of the parent node and its assigned nodes, and so on up to the root node.</td>
</tr>
<tr>
<td></td>
<td>● <strong>Downwards interlocking</strong> interlocks child nodes of the parent node, their child nodes, and so on down to the end nodes. This comprises a subhierarchy of interlocked nodes with the parent node at its root.</td>
</tr>
</tbody>
</table>

When applying the **Interlocking** setting, be aware of the following:

- Choosing the scope for hierarchy-specific changes involves striking a balance between centralized control and process efficiency.
- The Interlocking setting also defines the locking of nodes to avoid competing changes by multiple users who work on the hierarchy at the same time.

**Prerequisites**

To minimize business disruption, we recommend that you define the scope for changes to a hierarchy type when you define the hierarchy type within a data model. You can only change the scope for changes to a hierarchy type when no pending change requests exist for any hierarchy of this type. If you must change the scope after you have defined the hierarchy type and you must then transport your changes, ensure that no pending changes exist for the affected hierarchies in the target system.

**Example**

The hierarchy called Global consists of continents, countries, cities, and teams. A change request to add Rome as a child node to Italy as the parent node is pending. No other hierarchy-relevant change requests are pending. If you want to change nodes that are specified as **Interlocked** in the figures and descriptions below, you must use the pending change request that assigns Rome to Italy. For changes to other nodes, you can use separate change requests.

**Interlocking – Loose**

The figure below shows how the **Interlocking** setting of **Loose** affects a hierarchy in the scenario where Rome is added to Italy.
Loose interlocking affects all nodes that are assigned to the parent node of the node being changed. The node being changed is Rome and its parent node is Italy. Only the direct child nodes of Italy - Rome and Milan - are interlocked with the pending change request.

**Interlocking – Strict**

The figure below shows how the *Interlocking* setting of *Strict* affects a hierarchy in the scenario where Rome is assigned to Italy in a pending change request.
Upwards Interlocking
All nodes in the path from Italy to Global are interlocked. The child nodes of these nodes are also interlocked. Affected nodes include the following:

- Italy (parent node), Rome and Milan (child nodes)
- Europe (parent node of parent node), France and Italy (child nodes)
- Global (root node), Asia and Europe (child nodes)

Downwards Interlocking
All nodes in the subhierarchy below Italy are interlocked. Affected nodes include the following:

- Cities Rome and Milan, which are below country Italy (Also covered by upwards interlocking)
- Teams I and J, which are below city Rome
- Teams K and L, which are below city Milan
- Any other nodes that might be added in the future to any nodes descending from Italy

5.4.4 Navigation to the User Interface

In SAP Master Data Governance, you can navigate to user interfaces in the following ways:

- **Roles and Navigation** [external document]
  You use menu nodes provided in SAP NetWeaver Business Client to directly navigate to an application. The menu is defined using a PFCG role.

- **Object-and-Action Based Navigation** [page 115]
You navigate in this way to display or start the processing of master data objects.

- **Change-Request Based Navigation [page 116]**
  You navigate in this way to display a change request that contains one or more master data objects that are being processed.

### 5.4.4.1 Object-and-Action Based Navigation

You use object-and-action based navigation to display or to start the processing of master data objects.

The typical navigation sequence for the user is as follows:

1. **Launch the search application from the SAP NetWeaver Business Client (transaction NWBC).**
   - To do this, you can choose an entry from the role menu, click a link from the service map, or click a link from the home page.
2. **Search for an object.**
   - You search for a business object of interest and display the object in a new window. By clicking the link to display the business object, you initiate a logical action. Another possibility is choosing the **New** button. In all cases, the system opens a new window with the target UI.
3. **Process the object in the target user interface.**
   - Sometimes you can directly edit the object in the target user interface and sometimes you must choose the **Edit** button first. Choosing the **Edit** button triggers another logical action.

### Features

#### Navigation from NWBC to the Generic Search Application

The Web Dynpro application **Generic Search** (USMD_SEARCH) is available in the menu. The USMD_SEARCH node of the menu can include the object-independent configuration USMD_SEARCH or an object-specific configuration.

With the generic configuration, you can set default values for the data model and for the entity type, by specifying the following parameters in the menu node:

- **USMD_MODEL**
- **USMD_ENTITY**

The user can still override the defaults you set by selecting different options on the user interface.

If you specify a **Business Object Type** with the parameter **USMD_OTC**, users can no longer change the data model and the entity type. For more information on the required parameters, see *Configuration of the Generic Search* [external document].

#### Navigation to the Target UI

**Note**

In the following description, the **Generic Search** is used as an example of the **Current UI**.

If you do not specify the **USMD_OTC** parameter for the USMD_SEARCH Web Dynpro application, the system derives an **Business Object Type** for the data model and entity type based on Customizing.

The relevant settings are as follows:
• **Activity:** Customizing for *Master Data Governance, Central Governance* (transaction MDGIMG) under
  - [General Settings ▶ Data Modeling ▶ Edit Data Model]

• **View: Business Object Type**

When a user clicks a link to a business object, the system triggers logical action **DISPLAY**. Further logical actions are possible, for example **CREATE**. The availability of logical actions (defined in Customizing for *Master Data Governance* under [Process Modeling ▶ Business Activities ▶ Define Logical Actions]) depends on the UI, the UI configuration, and the state of the chosen business object type.

To determine the target user interface, the system considers the current user interface and its configuration, the logical action, and the **Business Object Type** of the selected business object. Relevant settings are available in the following Customizing activities:

• **Standard Definition:** Customizing for *Master Data Governance, Central Governance* (transaction MDGIMG)

• **Custom Definition (Overrides Standard Definition):** Customizing for *Master Data Governance, Central Governance* (transaction MDGIMG)

**Navigation from Other Search Applications**

You can use other search applications than **USMD_SEARCH**. In particular, the initial page of the single-object processing UI **USMD_OVP_GEN** can provide an object-specific search. The logic for determining a target UI is the same for other search applications as it is for the generic search.

**Navigation from the Initial Step of a Change Request**

If a logical action triggers the creation of a change request, the system gives precedence to the logic of the change-request-based navigation (see Change-Request Based Navigation [page 116]) over object-and-action based navigation for the initial change request step 00. If you have defined a target UI for the change request step 00, the system uses that UI. (Path: Customizing for *Master Data Governance, Central Governance* under [General Settings ▶ Process Modeling ▶ Change Requests ▶ Configure Properties of Change Request Step]). If you have not defined a target UI, the system uses the logic for object-and-action based navigation to determine a target UI.

**Note**

If the current UI application is the generic single-object processing UI **USMD_OVP_GEN** and the same UI is also the target UI application, the system ignores the value of **Configuration ID** from the view **User Interface per Change Request Step**. Instead, **USMD_OVP_GEN** uses its current configuration for the initial change request step.

### 5.4.4.2 Change-Request Based Navigation

Change-request based navigation occurs when you start processing a change request. Navigation possibilities include the following:

- Opening the relevant work item from the inbox.
- Searching for and selecting a change request in **My Change Requests**.
- Searching for a business object and clicking the **Pending Change Requests** icon.
Features

Navigation Based on Change Request Steps

The system selects a target user interface (UI) for the change request type and the current change step based on the User Interface per Change Request Step view of the following activity in Customizing for Master Data Governance, Central Governance: General Settings > Process Modeling > Change Requests > Configure Properties of Change Request Step. If a target UI is found, it is used.

Navigation Based on Change Request Type

If the system cannot identify a target UI for the current change request step, it selects a target UI for the change request type based on the Type of Change Request view of the following activity in Customizing for Master Data Governance, Central Governance: General Settings > Process Modeling > Change Requests > Create Change Request Type. If a target UI is found, it is used.

Object-and-Action Based Navigation for Single-Object Change Requests

If the system cannot identify a target UI for the current change request step and the indicator Single Object of the change request type is set (Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Change Requests > Create Change Request Type), the system uses object-and-action based navigation.

The system performs the following actions:

1. Determines the business activity for the change request type.
   
   **Path:** Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Change Requests > Create Change Request Type

2. Determines the logical action and Business Object Type based on the business activity settings.
   
   **Path:** Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Business Activities > Create Business Activity

3. Determines the target UI based on the business activity, logical action, and Business Object Type according to the logic of object-and-action based navigation (See Object-and-Action Based Navigation [page 115]).

**Note**

If the system cannot determine a target UI based on the logic described above, the target UI is the obsolete single-object processing UI USMD_ENTITY_VALUE2. In this case, the system determines the application configuration ID for USMD_ENTITY_VALUE2 in the Entity Types view of the following activity in Customizing for Master Data Governance, Central Governance: General Settings > Process Modeling > Change Requests > Create Change Request Type.

Multiple-Object Change Requests

If the system cannot determine a target UI for the current change request step and the Single Object setting of the change request type is not applied (Path: Customizing for Master Data Governance, Central Governance under General Settings > Process Modeling > Change Requests > Create Change Request Type), the target UI is application USMD_CREQUEST_PROCESS.
5.4.5 Enabling Detailed Analysis of Change Requests

You can apply system settings that allow you to monitor in detail how effectively your organization processes change requests. You can analyze the statuses of change requests in your organization, the processing times of change requests in your organization, and the nature of change requests involving you. For more information, see Analysis of Change Requests [external document].

Procedure

Enabling the detailed analysis of change requests involves completing the following tasks:

1. Configuring Operational Data Provisioning
2. Activating Business Information (BI) Content in Master Data Governance
3. Setting up the business context viewer
4. Assigning roles to your user
5. Changing authorization objects
6. Integrating SAP BusinessObjects Dashboards
7. (Optional) Defining a service-level agreement

Configuring Operational Data Provisioning

For more information, see Operational Data Provisioning [external document].

Activating BI Content in Master Data Governance

You use Business Information (BI) content to analyze change requests.

To activate the content, proceed as follows:

1. Run transaction BSANLY_BI_ACTIVATION.
2. Choose the OMDG_ANLY_CR_PROCESS content bundle.
3. Optional Step: If you want to identify and fix the errors that would occur if you activated the content bundle, choose the Simulate Activation button.
4. To activate the content bundle, choose the Activate button.

Setting Up the Business Context Viewer

You must activate the business context viewer to be able to access side panels for the following Web Dynpro applications that are used in the analysis of change requests:

- **Processing Time (List View)** (MDG_MONITOR_CR_PROCESTIME).
- **My Change Requests** (USMD_EDITION_CREQUEST)

You can refer to the following documents:

- For instructions on how to activate the business context viewer, see Business Context Viewer in Single Processing [external document].
- For more information about the business context viewer, see Business Context Viewer (BCV) [external document].
- For more information about side panels, see Side Panel [external document].
You can only access the side panels after you change the authorization object \textit{Business Context Viewer \textgreater Execute Side Panel (BCV\_SPANEL)}. Instructions on how to do this are provided in the \textit{Changing Authorization Objects} section of this document.

\textbf{Note}

After you activate the business context viewer, you can configure a side panel for any Web Dynpro application.

\section*{Assigning Roles to Your User}

You need to assign roles to your user. For more information, see \textit{Authorization Concept in Business Context Viewer (BCV)} [external document]

\section*{Roles to Access Web Dynpro Applications}

Investigate if the role or roles you already have allow you to access the following Web Dynpro applications:

\begin{table}[h]
\centering
\begin{tabular}{|l|l|}
\hline
\textbf{Web Dynpro Application} & \textbf{Description} \\
\hline
MDG\_MONITOR\_CR\_PROCETIME & Used for the analysis of the status of change requests or the processing time of change requests. \\
\hline
MDG\_ANLY\_CR\_REJ\_REASON & Used to display the reasons why change requests were rejected. \\
\hline
USMD\_EDITION\_CREQUEST & Used to display change requests involving you. \\
\hline
\end{tabular}
\end{table}

\textbf{Note}

You can view and edit roles using transaction \texttt{PFCG}. The \textit{Menu} tabbed page shows Web Dynpro applications. Often, existing roles that use the required Web Dynpro applications have technical names with suffixes of \texttt{*\_MENU}.

If you do not have the required roles, consider the following options:

\begin{itemize}
\item Assign the Master Data Governance: Analytics (\texttt{SAP\_MDGA\_MENU}) role to your user.
  
  This role only contains the relevant Web Dynpro applications.
\item Create your own role and add the Web Dynpro applications to that role.
  
  If you do this, you can control the placement of Web Dynpro applications on the menu in the user interface.
\end{itemize}

\section*{Changing Authorization Objects}

You must modify authorization objects to accomplish the following:

\begin{itemize}
\item Specify the change request types to be analyzed and the level of access required
\item Specify the Web Dynpro applications requiring a side panel.
\end{itemize}

For every role associated with the relevant Web Dynpro applications, proceed as follows:

1. Call up transaction \texttt{PFCG}.
2. Enter the name of the role and choose the \texttt{(Change) icon.}
3. Open the \textit{Authorizations} tab page and, in \textit{Maintain Authorization Data and Generate Profiles} section, choose the \texttt{(Change) icon.}
4. Change the relevant authorization objects as shown in the following table:
<table>
<thead>
<tr>
<th>Authorization Object</th>
<th>Purpose</th>
<th>Parameter</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP Master Data Governance ➤ Type of Change Request ➤ (USMD_CREQ)</td>
<td>Specify the types of change requests users are allowed to analyze and the level of access allowed.</td>
<td>Change Request Type</td>
<td>Specify the level of the access allowed to each the change request types specified under Activities. As a minimum, choose Display. Choose other options, if required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Activities</td>
<td>Specify which change request types can be accessed. You can use the * symbol as a wildcard for the entire technical name or for part of the technical name of the change request type.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Context Key</td>
<td>Enter the following context keys:</td>
</tr>
<tr>
<td>Business Context Viewer ➤ Execute Side Panel ➤ (BCV_SPANEL)</td>
<td>Specify the Web Dynpro applications requiring a side panel.</td>
<td></td>
<td>• MDGAF_MYCR</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The application is the Application Framework (MDGAF) and the object is My Change Requests (MYCR). Specifying this context key enables a side panel for the My Change Request screen.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• MDGAF_ANLY</td>
</tr>
</tbody>
</table>
| | | | The application is the Application Framework (MDGAF) and the object is (ANLY). Specifying this context key enables a side panel for the Status (Graphic...
<table>
<thead>
<tr>
<th>Authorization Object</th>
<th>Purpose</th>
<th>Parameter</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Front End Integration Xcelsius Dashboard (S_RS_XCLS)</td>
<td>Authorization for working with SAP BusinessObjects Dashboards.</td>
<td>RSXCLSID</td>
<td>Specify the technical name of the dashboard: 0XC_MDG_MONITOR_CR.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Activities</td>
<td>Specify the level of the access allowed to dashboards. As a minimum, choose Display. Choose other options, if required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RSZOWNER</td>
<td>Specify the owner of the dashboard for a reporting comment. We recommend a value of &quot;*&quot; to provide universal ownership.</td>
</tr>
</tbody>
</table>

5. Save the authorization profile and choose the (Generate Authorization Profile) icon.

**Integrating Dashboards**

For an overview of how to integrate dashboards, see *Xcelsius Enterprise Integration* [external document]

SAP Business Objects dashboards only work if a BI Java server is enabled. For more information, see SAP Note 1450981.

**(Optional) Defining a Service-Level Agreement**

A change request is late if it exceeds its due date (an optional field of the change request) or if it violates a Service Level Agreement (SLA). You can define the SLA in Customizing for priorities of change request types.

To define a service level agreement for each priority of a change request type, proceed as follows:

1. In Customizing for Master Data Governance, Central Governance, choose > General Settings > Process Modeling > Change Requests > Create Change Request Type
2. In the Type of Change Request view, choose a Change Request Type
3. In the Service Level Agreement view, define a target number of days and hours for each Priority.
   - When specifying hours, you can only specify 4 hours, which is a half day.

**Result**

After completing this procedure, it is possible to access meaningful analytical information about change requests.
5.5 Data Replication

You use this function to make the necessary settings for data replication within the Data Replication Framework (DRF). Data is replicated to business systems that are assigned to outbound implementations of replication models (see graphic below).

![Figure 18: Replication Models, Outbound Implementations, and Business Systems](image)

Features

SAP supplies most of the objects and assignments as standard, such as the following:

- Business objects
- Filter objects
- Outbound implementations
- Service operations

**Note**

You can view the objects and assignments that SAP supplies as standard in Customizing. More information is available in Customizing for *Master Data Governance, Central Governance* under **General Settings > Data Replication**.

However, you must configure your replication model. This involves the following:

- Definition of replication model
  - Define your replication model and enter a description.
- Definition of the receiving systems of your replication model
  - Enter one or more systems in which you want to replicate the data.
Caution
When you transfer your Customizing settings to the production system, you must subsequently check and adjust the assignment of the replication model to the receiving systems.

- Assignment of outbound implementations
  You assign the desired outbound implementations to your replication model.
  The outbound implementation specifies how business object data is transferred to a replication model. This means, for example, which data is to be transferred and which communication channel is to be used.

  Note
  The following communication channels are available for data replication purposes:
  - RFC
  - IDoc
  - Enterprise Services
  - File download
  Ensure that you use only those communication channels that are supported by your replication model.

One or more outbound implementations can be used for each replication model. This means that different data can be transferred in different ways for a certain replication model. This depends on the attributes of the outbound implementations that are assigned to the replication model.

- Definition of additional parameters per outbound implementation
  You can define technical parameters for parallel processing or specify the number of business objects per bulk message, for example.

- Assignment of languages
  You can specify which languages the system is to take into account for language-dependent texts in the outbound implementation.

Activities
You configure your replication model in Customizing for Master Data Governance, Central Governance under General Settings Data Replication Define Custom Settings for Data Replication Configure Replication Models.

For further prerequisites for data replication, see Customizing for Master Data Governance, Central Governance under General Settings Data Replication Overall Information.

5.5.1 Configuring Data Replication

You can replicate master data stored within MDG as well as reference data, stored in configuration tables. The replication process is slightly different in each case.

MDG offers the following options to store active master data (data that has been approved):
- The reuse option used by MDG-M and MDG-S stores data in the SAP ERP tables such as MARA or LFA1.
- The flex option used by MDG-F and MDG for Custom Objects stores data in generated tables.
In both options, inactive master data (data that has not yet been approved) is stored in the generated tables. Data that the MDG system replicates to target systems is always active data. The MDG system takes the active data from the SAP ERP tables or from the generated tables depending on the option in use (reuse option or flex option).

MDG applications such as MDG-M, MDG-S, and MDG-F include standard implementations of the Data Replication Framework (DRF) that read the data and send the messages to the target system. The standard implementations support key mapping and value mapping.

**About the Configuration of Reference Data**

Reference data, which is stored in Customizing tables, is typically stable and available for use across an organization. Currency codes, for example, may be stored as reference data. You can model reference data in MDG data models, govern changes, and replicate changes.

Once configured, the replication process for reference data is as follows:

- Replication from the MDG hub to Customizing tables in one or more target systems.
- Creation and release of transport request in target system.

**Prerequisites**

At least one data model, with entity types, attributes, and relationships is defined using the flex model. The user interface, workflow, and processors are defined.

Prerequisites for the replication of Reference data are as follows:

1. The target system of replication is a development system.
2. The user who replicates the data has the same ID in the source system and in the target system.
3. The RFC destination for the target system has the following settings:
   1. Under Logon & Security, you have selected Trust Relationship.
4. In the target system, the user who replicates the data is added to the list of RFC Users authorized to execute RFC calls in trusted systems.

**Procedure**

**Configuring Data Replication**

1. Define mapping contexts across clients for the Unified Key Mapping Service (UKMS).
   - **Path:** Customizing for Key Mapping (transaction IDMIMG) under Define a Mapping Context for UKMS Define Mapping Contexts.
   - **Instructions:** Copy the default Main Context to a new table and give the new mapping context a prefix of Z. The system generates a set of tables based on the standard tables. The Z prefix indicates that the objects in those tables belong to the Customer namespace.
   - **Example:** Copy table UKMDB_AGC00000 to ZUKMDB_AGCZZSF0.
2. Define the business object types to be replicated using outbound implementations and assign the defined business object types to a main context ID that is defined within the UKMS (Unified Key Mapping Service). Business object types used in data replication are based on entity types with a storage and use type of 1 within data models.
1. Define business object types to be replicated.
   **Path:** Customizing for *Key Mapping* (transaction IDMIMG) under *Define Business Objects*
   **Example:** Specify business object type ZZSF, which is for the customer implementation of SFLIGHT (Flex).

2. Assign the business object types you defined to the mapping context for key mapping.
   **Path:** Customizing for *Key Mapping* (transaction IDMIMG) under *Enhance Key Mapping Content: Assign Business Objects to the Main Context*
   **Example:** Assign business object type ZZSF to the main context (defined in the previous step.)

3. If multiple object identifier types that already belong to the same business object type must belong to the same mapping group, define an object node type. Then assign the object node type to the object identifier types.
   1. Define an object node type.
      **Path:** Customizing for *Data Replication* (transaction DRFIMG) under *Enhance Default Settings for Outbound Implementation: Define Business Objects and Object Identifiers: Define Business Object Nodes*
   2. Specify the object node type in the definitions of the object identifier types.
      **Path:** Customizing for *Key Mapping* under *Enhance Key Mapping Content: Define Object Identifiers.*
      - **Row for Business Partner Number**
        - **Business Object Type:** 147
        - **Business Object Node Type:** 368
        - **Business Object ID Type:** 888
        - **Constant Name for Business Object ID Type:** BPARTNER_UUID.
      - **Row for Business Partner UUID**
        Same as above, except for the following:
        - **Business Object ID Type:** 889
        - **Constant Name for Business Object ID Type:** BPARTNER_UUID.

3. Define business object identifier types so it is possible to differentiate an identifier of a business object from other identifiers of the same business object. Assign the business object identifiers types to the relevant business object types.
   1. Create business object identifier types.
      **Path:** Customizing for *Key Mapping* (transaction IDMIMG) under *Enhance Key Mapping Context: Define Object Identifiers.*
      **Example:** Object ID Type = ZZSF; Description of Object ID Type = SFLIGHT - Airline Code; BO Type = ZZSF; Ob ID Constant Name = ZZSF_AIRLCODE; Object Node Type = ZZSF
   2. Assign the business object identifier types to the definition of the business object types.
      **Path:** Customizing for *Key Mapping* (transaction IDMIMG) under *Enhance Key Mapping Context: Define Business Objects.*
      **Example:** Business Object Type = ZZSF; Description = SFLIGHT (Flex Option); Constant Name = ZZSF_AIRLCODE; Object Identifier Type for Key Structure Access = ZZSF
Steps for Replicating Data

1. Create a package in preparation for the generation of data model-specific structures in the ABAP Dictionary.
   Generate the structures for each entity type that you want to replicate. Verify that the system generated the
   structures correctly.
   1. Run transaction SE80 and create a package.
      Example: Object Name = ZZ_DRF, Description = Custom Object SFLIGHT Data Replication
   2. Generate the data model-specific structures.
      Path: Customizing for Master Data Governance, Central Governance (transaction MDGIMG) under
             Data Modeling ➤ Generate Data Model-Specific Structures
      Example: Data Type = ZXX_S_ZZ_ZDRF_CARR, Where Used = DRF Structures, Prefix / Namespace = ZXX, Name of Structure = ZDRF_CARR
   3. Run transaction SE11 and check the structures for the generated data model-specific structure.
      Whenever MDG generates structures, it activates them, so the word Active displays beside the structure
      name.
      Example: Data Type = ZXX_S_ZZ_ZDRF_CARR

2. Assign a key structure to the object identifier types so the key mapping functions can break down
   concatenated keys into their constituent parts.
   Path: Customizing for Data Replication (transaction DRFIMG) under
          Enhance Default Settings for Outbound Implementation ➤ Define Business Objects and Object Identifiers ➤ Assign Key Structures to Object
          Identifiers
   Example: Business Object Type = ZZSF; Key Structure = ZXX_S_ZZ_ZDRF_CARR.

3. Assign an entity type within a data model to a business object type, generate the data model, and verify that
   the confirmation message returns no errors.
   Path: Customizing for Master Data Governance, Central Governance (transaction MDGIMG) under
          General Settings ➤ Data Modeling ➤ Edit Data Model ➤ In the Inactive Data Models view, select a data model. In the
          Entity Types view, select an entity type. In the Business Object Type view, enter the business object type for
          the entity type. Then generate the data model.
   Example: In the Inactive Data Models view; select CARR. In the Entity Types view, select ZZ. In the Business
           Object Types view, enter the following details: Business Object Type = ZZSF. Choose the Generate Data Model
           button.

4. Prepare for the creation of an outbound interface model by creating a package that contains a function
   group.
   1. Run transaction SE80.
   2. Create a package.
      Example: Package = Z_ZZ_PACKAGE. Short Description = Package for Outbound Implementation
               for ZFLIGHT (ZZ). Software Component = HOME. Transport Layer = ZZNE. Package Type = Not a Main
               Package.
   3. Create a function group for the package.
      Example: Function Group = Z_ZZ_FUNC_GROUP. Short Text = Function Group for Outbound
               Sflight (ZZ)

5. Generate an outbound interface model that contains the entities and attributes from a data model that you
   want to replicate from the Master Data Governance hub to one or more target systems. This model also
   generates interfaces (RFCs and service interfaces) that can be used for such a data replication. After
   creating the outbound interface model, you can view the generated function module in transaction SE80.
**Path:** Transaction **OIF_MAINTAIN** or Customizing for **Data Replication** (transaction **DRFIMG**) under **Enhance Default Settings for Outbound Implementation** > **Define Outbound Interface Models**

**Example:** Complete the following steps:

1. **Wizard step:** **Enter Header Data.**
   
   Specify and describe an identifier for the interface model, an object type code, a package name, a name for the outbound interface model, and a description for the interface model. If you want to replicate reference data, so triggering the generation of a function module enabling the replication of such data, select the **Configuration Data** checkbox.

   **Note**
   
   You can adjust the generated function module according to your needs, for example, in the case of reference data, you can omit the release of the transport request in order to enable data enrichment. The user can then release the transports manually.

   **Example**
   
   Select the **CARR** entity type and all of its attributes. Choose the Name ABAP Dictionary Objects button. Enter the following details for the new structure:

   - **Interface Model ID** = **ZZ_SFLIGHT**
   - **Interface Model Description** = **SFlight Outbound Model (ZZ)**
   - **Object Type Code** = **ZZSF**
   - **Package Name** = **Z_ZZ_FUNC_GROUP**
   - **Name** = **ZZ_SFLIGHT**
   - **Description** = **Generated RFC for SFlight Outbound Model (ZZ)**

2. **Wizard step:** **Select Entity Types and Attributes.** Select entity types and attributes you want to include in the interface model. Then enter names for the resulting dictionary objects, by choosing the **Name ABAP Dictionary Objects** button.

   **Example:** Select the **CARR** entity type and all of its attributes. Choose the Name ABAP Dictionary Objects button. Enter the following details for the new structure:

   - **Structure Name** = **ZZSF_S_CARR**
   - **Structure Description** = **Structure for CARR**
   - **Table Type Name** = **ZZSF_T_CARR**
   - **Table Type Description** = **ZZSF_T_CARR**

3. **Wizard Step:** **Review and Submit.** Review and submit your work. Create a transport request or assign an existing transport request. You can use the same transport to transfer the function module to the target system later on.

4. **Wizard Step:** **Check Application Log.** Check the application log that displays after you review your submitted work.

5. Review the code of the function module. Run transaction **SE80**. Open the **Repository Browser** and browse by the **Function Group** you created earlier. Open the **Function Modules** folder, and review the system-generated function module for the outbound interface model.

   The outbound implementation you define in the data replication framework calls this function module to replicate the data.
6. Create an outbound implementation to define how specific business object data is replicated. The creation of the outbound implementation involves specifying business object data to be transmitted, a class that retrieves and sends the data, and a communication channel.

When defining an outbound implementation, use the generic outbound implementation class *(CL_MDG_OIF_DRF_OUTBOUND_IMPL)*. You can copy this class to allow additional capabilities that are not supported by default such as key mapping and value mapping. For more information, you can refer to the standard outbound implementations that SAP delivers for other objects.

**Path:** Customizing for Data Replication (transaction DRFIMG) under Enhance Default Settings for Outbound Implementation ▶ Define Outbound Implementations ▶

**Example:** *Outbound Implementation = ZZSF_01; Outbound Implementation Class = CL_MDG_OIF_DRF_OUTBOUND_IMPL; Communication Channel = 4 Replication via iDoc; Business Object Type = ZZSF; Outbound Interface Model ID = ZZ_SFLIGHT.*

7. Create a filter object to restrict the data can be selected and transferred to a target system during data replication through the use of filters. Define filters for the filter object.

**Path:** Customizing for Data Replication (transaction DRFIMG) under Enhance Default Settings for Outbound Implementation ▶ Define Filter Objects ▶

1. Enter data in the Define Filter Objects view and select the relevant row.

**Example:** *Filter Object = ZZSF_FROOT; Description = Filter SFlight (ZZ) - Root. Leave the Table Name field blank.*

**Note**
A complex filter such as the one in the example does not require a table name. The system only requires table names for simple filters. Such filters are only available for standard applications that are built using the reuse option.

2. Define the filters for the filter object in the Assign Filters subview.

If required, you can define your own structure to include all relevant fields from the generated table. In the Assign Filters view, apply the following settings.

1. For the Filter field, use codes between 80 and 99. This range is assigned to the customer namespace.
2. Specify a filter class. **Example:** Use the generic Filter Class CL_MDG_OIF_DRF_FILTER.

8. Assign a filter object to a business object type (specific filtering) or to an outbound implementation.

- Assignment of a filter object to a business object type (specific filtering).
  **Path:** Customizing for Data Replication (transaction MDGIMG) under Enhance Default Settings for Outbound Implementations ▶ Define Business Objects and Object Identifiers ▶ Assign Filter Objects to Business Objects ▶

- Assignment of a filter object to an outbound implementation:
  **Path:** Customizing for Data Replication (transaction MDGIMG) under Enhance Default Settings for Outbound Implementations ▶ Define Outbound Implementations ▶

**Example:** *Business Object Type = ZZSF; Filter Object = ZZSF_FROOT; Outbound Interface Model ID = ZZ_SFLIGHT*

9. Create a filter to indicate precisely what data you want to replicate.

1. Run transaction DRFF.
2. Select the Business Object for which you want to define filter criteria.
Example: SFLIGHT (Flex Option).

3. Define a filter.
   Example: Under Filter Criteria to Include Business Objects, choose Airline local currency is EUR.

10. Create a replication model, assign the outbound implementation to the replication model, and assign the business systems that act as target systems for replication to the combination of the outbound implementation and the replication model. Each replication model specifies one or more outbound implementations.

1. Create a replication model.
   Client-Specific Path: Customizing for Data Replication (transaction MDGIMG) under Define Custom Settings for Outbound Implementations ➔ Define Replication Models
   Example:
   In the Define Replication Model view, create a new entry with the following settings:
   Replication Model = ZZSF; Description = Replication Model for SFLIGHT; Log - Days: 15.
   Select the new entry.

2. Assign an outbound implementation to the replication model.
   Example: In the Assign Outbound Implementation view, apply the following settings: Outbound Implementation = ZZSF_01; Communication Channel = 4 Replication via RFC; Filter Time = 2 Filter After Change Analysis.
   Select the assigned Outbound Implementation.

3. Assign the business system or business systems to which you want to replicate data using the combination of the replication model and the outbound implementation.
   Example: Open the Assign Receiver Systems view, and enter the following value: Business System = QV5_410

4. Activate the replication model.
   Choose the Activate Replication Model pushbutton.

Additional Steps for Replication of Reference Data

1. Generate an outbound interface model that contains the entities and attributes from a data model that you want to replicate from the Master Data Governance hub to one or more target systems. This model also generates interfaces (RFCs and service interfaces) that can be used for such a data replication. After creating the outbound interface model, you can view the generated function module in transaction SE80.
   Path: Transaction OIF_MAINTAIN or Customizing for Data Replication (transaction DRFIMG) under Enhance Default Settings for Outbound Implementation ➔ Define Outbound Interface Models
   Example: Complete the wizard as follows:
   1. Wizard step: Enter Header Data.
      Specify and describe an identifier for the interface model, an object type code, a package name, a name for the outbound interface model, and a description for the interface model. To trigger the generation of a function module enabling the replication of such data, select the Configuration Data checkbox.

   Note
   You can adjust the generated function module according to your needs, for example you can omit the release of the transport request in order to enable data enrichment. The user can then release the transports manually.
Example
Select the CARR entity type and all of its attributes. Choose the Name ABAP Dictionary Objects button. Enter the following details for the new structure:

- **Interface Model ID** = ZZ_SFLIGHT
- **Interface Model Description** = SFlight Outbound Model (ZZ)
- **Object Type Code** = ZZSF
- **Package Name** = Z_ZZ_FUNC_GROUP
- **Name** = ZZ_SFLIGHT
- **Description** = Generated RFC for SFlight Outbound Model (ZZ)

2. **Wizard step: Select Entity Types and Attributes.** Select entity types and attributes you want to include in the interface model. Then enter names for the resulting dictionary objects, by choosing the Name ABAP Dictionary Objects button.

   **Example:** Select the CARR entity type and all of its attributes. Choose the Name ABAP Dictionary Objects pushbutton. Enter the following details for the new structure:
   - **Structure Name** = ZZSF_S_CARR
   - **Structure Description** = Structure for CARR
   - **Table Type Name** = ZZSF_T_CARR
   - **Table Type Description** = ZZSF_T_CARR

3. **Wizard Step: Review and Submit.** Review and submit your work. Create a transport request or assign an existing transport request. You can use the same transport to transfer the function module to the target system later on.

4. **Wizard Step: Check Application Log.** Check the application log that displays after you review your submitted work.

5. Review the code of the function module. Run transaction SE80. Open the Repository Browser and browse by the Function Group you created earlier. Open the Function Modules folder, and review the system-generated function module for the outbound interface model.

   The outbound implementation you define in the data replication framework calls this function module to replicate the data.

2. Create a mapping using the service mapping tool to map data from the staging area in the source system to the reuse table in the target system.

   Ensure the following:
   - The source structure is the data model-specific structure for the outbound interface model.
   - The source structure uses the flex option and the target structure uses the reuse option.

   **Path:** Customizing for Master Data Governance, Central Governance (transaction MDIMG) under Data Modeling - Generate Data Model-Specific Structures

3. Maintain a mapping table to map the tables to the objects.

   Run transaction SM30.

   **Example:**
   - **Table Name:** ZFX_S_ZT024E. Specified when you created an outbound interface model.
   - **Object:** V_T024E. The target business object stored in Customizing.
5.5.2 Data Replication Using Enterprise Service Oriented Architecture (SOA)

The replication of master data from the MDG hub to the connected systems and clients can be done using Enterprise Service Oriented Architecture (SOA).

The service interfaces and service groups to be configured are found in the SOAMANAGER (SOA Manager) transaction.

For detailed information how to configure the SOAMANAGER for MDG, see Configuring the SOA Manager for Master Data Governance (NW 7.40) [external document] (NetWeaver 7.40 or higher).

For information on the SOAMANAGER steps to be performed to configure WebService-based communication, see Configuring a Consumer Proxy [external document].

The following services need to be configured:

**Note**
Service definitions and service groups that need to be configured additionally to run MDG-C on a client system are shown in brackets.

- For the message inbound processing in the master data client systems, configure the following services:
  - BusinessPartnerSUITEBulkReplicateRequest_In
  - BusinessPartnerRelationshipSUITEBulkReplicateRequest_In
  - [BusinessPartnerSUITEBulkReplicateConfirmation_In]
  - [BusinessPartnerRelationshipSUITEBulkReplicateConfirmation_In]

- Configure the following service groups:
  - MDG_BS_SUPPLIERREPLICATECONF
  - [MDG_BS_SUPPLIERREPLICATEREQ]

- For the message inbound processing in the master data server system, configure the following services:
  - BusinessPartnerSUITEBulkReplicateConfirmation_In
  - BusinessPartnerRelationshipSUITEBulkReplicateConfirmation_In
  - [BusinessPartnerSUITEBulkReplicateRequest_In]
  - [BusinessPartnerRelationshipSUITEBulkReplicateRequest_In]

- Configure the following service groups:
  - MDG_BS_SUPPLIERREPLICATEREQ
  - [MDG_BS_SUPPLIERREPLICATECONF]

Replication eSOA Using the Data Replication Framework (DRF)
Prerequisite: Receiving systems/clients are defined in the SLD (system landscape directory). For more information, see Customizing for Master Data Governance, Central Governance under General Settings Data Replication Overall Information.
1. Define technical settings for business system under General Settings » Data Replication » Define Technical Settings for Data Replication » Define Technical Settings for Business Systems

2. Enter a business system, for example ABC_123 and a logical system, for example ABCCLNT123.

2. Define replication model (replication of business partner) under General Settings » Data Replication » Define Custom Settings for Data Replication » Define Replication Models

1. Using New Entries, define a new replication model and enter a description.

2. Mark the line and choose Assign Outbound Implementation. Create a new entry and enter the following values using the input help:
   - Outbound Implementation: 986_3 Outbound Impl. for BP/REL via Services
   - Communication Channel: Replication via Services
   - Filter time: Filter After Change Analysis

2. Assign Target Systems for Repl. Model /Outb.Impl. Create a new entry and enter the business system name for the receiving system created in the step before.

4. Choose Assign Outbound Parameter. Create a new entry, enter the following values, and save.
   - Outb. Parameter: PACK_SIZE_BULK
   - Outb. Parameter value: 100 (example value)

5. Return to the Define Replication Model view. Optionally, you can also add an expiration time for the log.

6. Save your replication model.

7. Mark the newly-created replication model, and choose Activate.

For more information, see Master Data Governance, Central Governance under General Settings » Data Replication » Define Custom Settings for Data Replication » Define Replication Models.

5.5.3 Data Replication of Customer Master Data Using ALE

Replication of master data from the MDG hub to the connected systems and clients can be done using Application Link Enabling (ALE).

1. Set up RFC connections (in both directions: hub to client and client to hub for ADRMAS and DEBMAS)

   1. Set up the RFC connection: Run transaction SM59 (configuration of RFC connections) and provide the required RFC destination details.

   2. To define a logical system in Customizing for SAP NetWeaver run transaction SALE and then choose Basic Settings » Logical Systems » Define Logical System. Enter all target systems/clients as logical systems.

2. Create Global Company code for ALE

   Run this activity in Customizing for SAP NetWeaver under Application Server » iDoc Interface / Application Link Enabling (ALE) » Modelling and Implementing Business Processes » Global Organization Units » Cross-System Company Codes. Create cross system company codes and map all company codes in use to the global company code defined.

3. Maintain distribution model in SAP ERP (Outbound) for the distribution of Customer Master using ADRMAS and DEBMAS:
1. Run transaction `SALE` (Display ALE Customizing) and choose [Modeling and Implementing Business Processes › Maintain Distribution Model and Distribute Views](#) Alternatively, run transaction `BD64` (Display Distribution Model).

2. In change mode, create a new model. Choose the [Create Model View](#). Enter a short text and a technical name.

3. Choose [Add Message Type](#) for the newly created model.

4. Enter names for the logical source and destination systems and choose message type `DEBMAS`.

5. Choose [Add BAPI](#) for the newly created model.

6. Enter the logical sender system and receiver system. Enter the following values:

<table>
<thead>
<tr>
<th>Obj.Name/Interface</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddressOrg</td>
<td>SaveReplica</td>
</tr>
<tr>
<td>AddressPers</td>
<td>SaveReplica</td>
</tr>
<tr>
<td>AddressContPart</td>
<td>SaveReplica</td>
</tr>
</tbody>
</table>

7. Save your changes.

4. Create partner profile

1. Run transaction `SALE` and choose [Modeling and Implementing Business Processes › Partner Profiles › Generate Partner Profiles](#) Alternatively, run transaction `BD82` (Generate Partner Profile).

2. Select the newly created model and in the [Partner System](#) field, enter the logical destination system.

3. Enter the ALE-User and the following values in the corresponding fields, and execute.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>3</td>
</tr>
<tr>
<td>Pack Size</td>
<td>100</td>
</tr>
<tr>
<td>Output Mode</td>
<td>Transfer IDoc immediately</td>
</tr>
<tr>
<td>Inb. Parameters: Processing</td>
<td>Trigger immediately</td>
</tr>
</tbody>
</table>

4. To verify your settings, run transaction `WE20` and from the [Partner Profiles](#) menu, choose [Partner type LS](#). Verify that [Partner type LS](#) is the logical destination system.

5. In the detail screen, the chosen message types `ADRMAS, ADR2MAS, ADR3MAS` and `DEBMAS` must appear as outbound parameters.

5. Distribute model view to receiving system.

1. Run transaction `SALE` and choose [Modeling and Implementing Business Processes › Maintain Distribution Model and Distribute Views](#) Alternatively, run transaction `BD64`.

2. Select the new model and choose [Edit › Model view › Distribute](#)

3. Verify that the correct receiving system is marked and choose [Enter](#).

4. Verify within the receiving system that the model view was created.

1. Run transaction `SALE` and choose [Modelling and Implementing Business Processes › Partner Profiles › Generate Partner Profiles](#) Alternatively, run transaction `BD82`.

2. Select the distributed model and the partner system.

3. Enter the ALE-User and the following values in the corresponding fields, and execute.
4. To verify your settings, run transaction WE 20 and from the partner profiles menu, choose partner type LS. Verify that partner type LS is the logical destination system.

5. In the detail screen, the chosen message types ADRMAS and DEBMAS must appear as inbound parameters. Double-click the message type to ensure that for ADRMAS, ADR2MAS and ADR3MAS, the process code BAPI is selected, and that for DEBMAS, the process code DEBM is selected. If any other process codes are selected, change them to the above-mentioned ones before proceeding.

6. Replication ALE Using the Data Replication Framework (DRF)

Prerequisite: Receiving systems/clients are defined in the SLD (system landscape directory). For more information, see Customizing for Master Data Governance, Central Governance under General Settings Data Replication Overall Information.


2. Enter a business system, for example ABC_123 and a logical system, for example ABCCLNT123, select the Define Bus. Systems, BOs view, and enter the BO Type 159 (Customer).

3. Make sure that under General Settings Data Replication Enhance Default Settings for Outbound Implementations Define Outbound Implementations in the line 986_5 (MDG Customer via ALE) the Sup. Key H (Support Key Harmonization) flag is set.

4. Mark the BO Type 159 (Customer), select the Define Bus. Systems, BOs, Communication Channel view, enter the C. Channel Replication via IDoc, and select the value Key Mapping for the Key Harm. field.

Note
If you select Harmonized IDs (no key mapping) for Key Harm., you have to use ALE inbound. It is not possible to import data using MDG DIF (Data Import Framework).

2. Define replication model (replication of customer) under General Settings Data Replication Define Custom Settings for Data Replication Define Replication Models.

1. Using New Entries, define a new replication model and enter a description.

2. Mark the line and chooseAssign Outbound Implementation. Create a new entry and enter the following values:
   - Outb. Impl.: 986_5 (MDG Customer via ALE)
   - Communication Channel: Replication via IDoc
   - Filter time: 2 (filter after change analysis)

3. Mark the line and chooseAssign Target System for Repl. Model/Outb.Impl. Create a new entry and enter the business system name for the receiving system created in the step before.
4. Choose **Assign Outbound Parameter**. Create a new entry, enter the following values, and save.
   ○ Outb. Parameter: PACK_SIZE_BULK
   ○ Outb. Parameter value: 100 (example value)

5. Return to the **Define Replication Model** view. Optionally, you can also add an expiration time for the log.

6. Choose **Save** to save your entries.

7. Mark the newly created replication model, and choose **Activate**.

   For more information, see **Master Data Governance, Central Governance** under **General Settings > Data Replication > Define Custom Settings for Data Replication > Define Replication Models**

7. **Enable monitoring of IDoc posting (ALE Audit)**
   1. In the hub system: Run transaction **BD64** to maintain a distribution model and define the direction MDG client to MDG hub.
      
      For more information about distribution models, see **Distribution Model** [external document].
   2. Open transaction **WE20** and navigate to the partner profile for the client system.
   3. Enter the following inbound parameters without changing further data:
      ○ Message type: ALEAUD
      ○ Process code: AUD2

4. Replicate the distribution model to the target system using transaction **BD64**.

5. In the client system: To generate a partner profile open transaction **SALE** and choose **System Monitoring > IDoc Confirmation in Receiving System (ALE Audit) > Distribution Model for ALE Audit**. Then select **Environment > Generate Partner Profiles**.

6. Open transaction **WE20**, navigate to the partner profile and enter the outbound parameter ALEAUD.

7. To configure the replication of audit documents open the transaction **SALE** and choose **System Monitoring > IDoc Confirmation in Receiving System (ALE Audit) > Confirmation of Audit Data > Define Variant**.

   You can schedule the report RBDSTATE as a background job to run on a regular basis.

8. **Define IDoc serialization for addresses**

   For the replication of addresses into an ERP system, the message types DEBMAS and ADRMAS need to be serialized. Carry out the following steps to set the background processing for the message types:
   1. Call transaction **WE20**.
   2. Go to folder **Partner Type LS** and mark the profile of the MDG hub client.
   3. In the list of inbound parameters, double-click the message type ADRMAS.
   4. Select the radio button for **Processing by background program**.
   5. Save your changes.
   6. Repeat this procedure for the message types ADR2MAS and ADR3MAS.

9. **Define serialization groups for inbound processing (V TBD41)**

   To define the serialization groups for inbound processing in the ERP client, go to **SAP NetWeaver > Application Server > IDoc Interface / Application Link Enabling (ALE) > Modelling and Implementing Business Processes > Master Data Distribution > Serialization for Sending and Receiving Data > Serialization Using Message Types > Define Inbound Processing** and carry out the following steps:
   1. Choose **New Entries**.
   2. Enter the following values with your hub client ID as the sending system:
3. **Save your entries.**

You can schedule the report `RBDSER04` as a background job to run on a regular basis.

### 5.5.4 Data Replication of Business Partner Master Data Using ALE

Replication of business partner master data from the MDG hub to the connected systems and clients can be done using Application Link Enabling (ALE).

1. **Set up RFC connections** (in both directions: hub to client and client to hub)
   1. Set up the RFC connection: Run transaction `SM59` (configuration of RFC connections) and provide the required RFC destination details.
   2. To define a logical system in Customizing for SAP Netweaver run transaction `SALE` and then choose `Basic Settings > Logical Systems > Define Logical System`. Enter all target systems/clients as logical systems.

2. **Maintain distribution model in SAP ERP (Outbound) for the distribution of Business Partner Master**
   1. Run transaction `SALE` (Display ALE Customizing) and choose `Modeling and Implementing Business Processes > Maintain Distribution Model and Distribute Views`. Alternatively, run transaction `BD64` (Display Distribution Model).
   2. In change mode, create a new model. Choose the `Create Model View`. Enter a short text and a technical name.
   3. Choose `Add Message Type` for the newly created model.
   4. Enter names for the logical source and destination systems and choose message type `BUPA_INBOUND_MAIN_SAVE_M` and `BUPA_INBOUND_REL_SAVE_M`.

    **Note**

    For the maintenance of the distribution model in an SRM or a CRM system (Inbound), repeat steps 1 to 4 in the corresponding system.

    **Note**

    To distribute business partner data and business partner relationship data, configure your target system as described:
   1. Define a serialization group.
   2. Assign the message types listed below to the serialization group and enter the corresponding sequence numbers:

   Table 56
<table>
<thead>
<tr>
<th>Message Type</th>
<th>Seq. number</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUPA_INBOUND_MAIN_SAVE_M</td>
<td>1</td>
</tr>
<tr>
<td>BUPA_INBOUND_REL_SAVE_M</td>
<td>2</td>
</tr>
</tbody>
</table>

2. Run transaction SALE and choose Modelling and Implementing Business Processes > Master Data Distribution > Serialization for Sending and Receiving Data > Serialization Using Message Types > Define Inbound Processing and enter the following data:

   Table 57
<table>
<thead>
<tr>
<th>Group</th>
<th>Message Type</th>
<th>Sending system</th>
<th>Obj/Proc</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Serialization group]</td>
<td>BUPA_INBOUND_MAIN_SAVE_M</td>
<td>[Sending system]</td>
<td>[value, e.g. 100]</td>
</tr>
<tr>
<td>[Serialization group]</td>
<td>BUPA_INBOUND_REL_SAVE_M</td>
<td>[Sending system]</td>
<td>[value, e.g. 100]</td>
</tr>
</tbody>
</table>

3. Make sure that incoming IDocs are not processed immediately. Run transaction WE20 and choose Partner > Partner Type LS > [Sending system]. For both inbound parameters proceed as described:
   1. Select the inbound parameter.
   2. Choose DetailScreenInboundParameter.
   3. On the Inbound options tab, select Trigger by background program.
   4. Run transaction SE38 and create a variant for the Report RBDSER04. As Serialization Group enter the serialization group created above.

3. Create partner profile
   1. Run transaction SALE and choose Modeling and Implementing Business Processes > Partner Profiles > Generate Partner Profiles. Alternatively, run transaction BD82 (Generate Partner Profile).
   2. Select the newly created model and in the Partner System field, enter the logical destination system.
   3. Enter the ALE-User and the following values in the corresponding fields, and execute.

   Table 58
<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>3</td>
</tr>
<tr>
<td>Pack Size</td>
<td>100</td>
</tr>
<tr>
<td>Output Mode</td>
<td>Transfer IDoc immediately</td>
</tr>
<tr>
<td>Inb. Parameters: Processing</td>
<td>Trigger immediately</td>
</tr>
</tbody>
</table>

4. To verify your settings, run transaction WE20 and from the Partner Profiles menu, choose Partner type LS. Verify that Partner type LS is the logical destination system.
5. In the detail screen, the chosen message types BUPA_INBOUND_MAIN_SAVE_M and BUPA_INBOUND_REL_SAVE_M must appear as outbound parameters.

4. Distribute model view to receiving system.
   1. Run transaction SALE and choose [Modelling and Implementing Business Processes] [Maintain Distribution Model and Distribute Views]. Alternatively, run transaction BD64.
   2. Select the new model and choose [Edit] [Model view] [Distribute].
   3. Verify that the correct receiving system is marked and choose Enter.
   4. Verify within the receiving system that the model view was created.
      1. Run transaction SALE and choose [Modelling and Implementing Business Processes] [Partner Profiles] [Generate Partner Profiles]. Alternatively, run transaction BD82.
      2. Select the distributed model and the partner system.
      3. Enter the ALE-User and the following values in the corresponding fields, and execute.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>3</td>
</tr>
<tr>
<td>Pack Size</td>
<td>100</td>
</tr>
<tr>
<td>Output Mode</td>
<td>Transfer IDoc immediately</td>
</tr>
<tr>
<td>Inb. Parameters: Processing</td>
<td>Trigger immediately</td>
</tr>
</tbody>
</table>

4. To verify your settings, run transaction WE 20 and from the partner profiles menu, choose partner type LS. Verify that partner type LS is the logical destination system.

5. In the detail screen, the chosen message types BUPA_INBOUND_MAIN_SAVE_M and BUPA_INBOUND_REL_SAVE_M must appear as inbound parameters.

5. Replication ALE Using the Data Replication Framework (DRF)

   Prerequisite: Receiving systems/clients are defined in the SLD (system landscape directory). For more information, see Customizing for Master Data Governance, Central Governance under [General Settings] [Data Replication] [Overall Information].

   1. Define technical settings for business system under [General Settings] [Data Replication] [Define Custom Settings for Data Replication] [Define Technical Settings] [Define Technical Settings for Business Systems].

   1. To define a new business system and to maintain the logical system for the receiving systems, choose New Entries.
   2. Enter a business system, for example ABC_123 and a logical system, for example ABCCLNT123, select the Define Bus. Systems, BOs view, and enter the BO Type 147 (Business Partner) and 1405 (Business Partner Relationship).
   3. Make sure that under [General Settings] [Data Replication] [Enhance Default Settings for Outbound Implementations] [Define Outbound Implementations] in the line 986_1 (Outbound Impl. for MDG BP via ALE) the Sup. Key H (Support Key Harmonization) flag is set.
   4. Mark the BO Type 147 (Business Partner) and 1405 (Business Partner Relationship), select the Define Bus. Systems, BOs, Communication Channel view, enter the C. Channel Replication via IDoc, and select the value Key Mapping for the Key Harm. field.
2. Define replication model (replication of business partners) under [General Settings] > [Data Replication] > [Define Custom Settings for Data Replication] > [Define Replication Models]
   1. Using [New Entries], define a new replication model and enter a description.
   2. Mark the line and choose [Assign Outbound Implementation]. Create a new entry and enter the following values:
      ○ Outb. Impl.: 986_1 (Outbound Impl. for MDG BP via ALE)
      ○ Communication Channel: Replication via IDoc
      ○ Filter time: 2 (filter after change analysis)
   3. Mark the line and choose [Assign Target System for Repl. Model/Outb.Impl]. Create a new entry and enter the business system name for the receiving system created in the step before.
   4. Choose [Assign Outbound Parameter]. Create a new entry, enter the following values, and save.
      ○ Outb. Parameter: PACK_SIZE_BULK
      ○ Outb. Parameter value: 100 (example value)
   5. Return to the [Define Replication Model] view. Optionally, you can also add an expiration time for the log.
   6. Choose [Save] to save your entries.
   7. Mark the newly created replication model, and choose [Activate].

For more information, see [Master Data Governance, Central Governance] under [General Settings] > [Data Replication] > [Define Custom Settings for Data Replication] > [Define Replication Models].

8. Define filter criteria.
   In case different business partners are sent to receiver systems do the following:
   1. Select your replication model and choose [Create].
   2. Enter filter values to determine which business partners shall be distributed, or which won’t, for example, due to number range settings for business partner creation.

   Run transaction [DRFF] or in SAP NetWeaver Portal or SAP NetWeaver Business Client, choose [Data Exchange] > [Data Replication] > [Define Filter Criteria].

6. Enable monitoring of IDoc posting (ALE Audit)
   1. In the hub system: Run transaction [BD64] to maintain a distribution model and define the direction MDG client to MDG hub.
      For more information about distribution models, see [Distribution Model] [external document].
   2. Open transaction [WE20] and navigate to the partner profile for the client system.
   3. Enter the following inbound parameters without changing further data:
      ○ Message type: [ALEAUD]
4. Replicate the distribution model to the target system using transaction **BD64**.
5. In the client system: To generate a partner profile open transaction **SALE** and choose **System Monitoring > iDoc Confirmation in Receiving System (ALE Audit) > Distribution Model for ALE Audit**. Then select **Environment > Generate Partner Profiles**.
6. Open transaction **WE20**, navigate to the partner profile and enter the outbound parameter **ALEAUD**.
7. To configure the replication of audit documents open the transaction **SALE** and choose **System Monitoring > iDoc Confirmation in Receiving System (ALE Audit) > Confirmation of Audit Data > Define Variant**.
   You can schedule the report RBDSTATE as a background job to run on a regular base.

### 5.5.5 Configuration for Data Replication to CRM Systems

You have to set up a mapping for the attributes **Sales Group** and **Sales Office**, as these attributes in the CRM client systems are part of the **Organizational Management** while in the MDG hub system they are based on Customizing tables.

**Example**

In the MDG hub system your set of data looks as follows:

<table>
<thead>
<tr>
<th>Sales Organization</th>
<th>Distribution Channel</th>
<th>Division</th>
<th>Sales Office</th>
<th>Sales Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>01</td>
<td>01</td>
<td>0001</td>
<td>01</td>
</tr>
<tr>
<td>0001</td>
<td>01</td>
<td>01</td>
<td>0001</td>
<td>02</td>
</tr>
<tr>
<td>0001</td>
<td>01</td>
<td>01</td>
<td>0002</td>
<td>02</td>
</tr>
<tr>
<td>0002</td>
<td>01</td>
<td>01</td>
<td>0001</td>
<td>01</td>
</tr>
<tr>
<td>0002</td>
<td>01</td>
<td>01</td>
<td>0001</td>
<td>02</td>
</tr>
</tbody>
</table>

In the CRM client system your set of data looks as follows:

- Sales Organization O 50000001 with Distribution Channel 01 and Division 01
  - Sales Office O 50000002
    - Sales Group O 50000003
    - Sales Group O 50000050
  - Sales Office O 50000026
    - Sales Group O 50000027
- Sales Organization O 50000004 with Distribution Channel 01 and Division 01
  - Sales Office O 50000005
  - Sales Group O 50000006
  - Sales Group O 50000075

To map these data proceed as described below:
**Settings in the MDG hub system**

**Prerequisite**

Technical settings for business systems are defined in the Customizing activity under **Master Data Governance, Central Governance > General Settings > Value Mapping > Define Technical Settings for Business Systems**.

**Procedure**

- **Maintain Value Mapping**
  1. Run the Customizing activity **Master Data Governance, Central Governance > General Settings > Value Mapping > Maintain Value Mapping**.
  2. Mark **MDG_FND_PARTY_ROLE_CODE** and choose (Display mapping relationships).
    - Under **Assign Code Lists** enter a **Mapping ID** and select a **List Agency ID** and a **List ID** using the entry help.
    - Under **Define Value Mapping** map the **Internal Code Value** with the **External Code Value**.
  3. Mark **VKBUR** and choose **Display mapping relationships**.
    - Under **Assign Code Lists** enter the following data:
      
      | Mapping ID | List Agency ID     | List ID | List Version ID | No Mapping |
      |------------|--------------------|--------|----------------|------------|
      | 1          | <CRM client system>| VKBUR  | 1              | X          |
      | 2          | <CRM client system>| VKBUR  | 2              | X          |

  4. Mark **VKGRP** and choose **Display mapping relationships**.
    - Under **Assign Code Lists** enter the following data:
      
      | Mapping ID | List Agency ID | List ID | List Version ID | No Mapping |
      |------------|---------------|--------|----------------|------------|
      | 1          | <CRM client system> | VKGRP  | 1              | X          |
      | 2          | <CRM client system> | VKGRP  | 2              | X          |
      | 3          | <CRM client system> | VKGRP  | 3              | X          |

- **Assign Code Lists to Elements and Systems**
  1. Run the Customizing activity **Master Data Governance, Central Governance > General Settings > Value Mapping > Assign Code Lists to Elements and Systems**.
  2. Select the **Type Data Element** and the **Global Data Type** MDG_FND_PARTY_ROLE_CODE and enter a **Business System**, a **List ID**, and a **List Agency ID** using the entry help.

- **Create and Edit Key Mapping**
  1. Run the transaction **NWBC** and launch the corresponding NetWeaver Business Client.
  2. Choose **Data Exchange > Data Replication > Create and Edit Key Mapping**.
  3. Under **Object Selection** enter the following:
4. Choose **Add Row**.

5. In row 2 and subsequent enter the **System ID** of the CRM client system, select the **Business Object Type** **Organizational Center** and enter the corresponding **Object ID**.

6. Repeat the steps 3 to 5 for all the sales organizations in the MDG hub system.

- **Set up Integration with Business Partner in CRM**
  1. Run the Customizing activity **Master Data Governance. Central Governance → Master Data Governance for Customer → Set up Integration with Business Partner in CRM**.
  2. Select the CRM client system using the input help and activate mapping of tax classifications and the filter for reflexive partner roles.

### Settings in the ERP client system

To make the CRM client system known to the ERP client system, which is necessary for the creation of mapping table entries, run transaction SM34, enter **MDGVC_BUS_SYS** choose **Maintain** and enter the corresponding data and set the **Disabled for Replication** flag.

### Settings in the CRM client system

**Prerequisite**

To make the MDG hub system known to the CRM client system run transaction **SM34**, enter **MDGVC_BUS_SYS** choose **Maintain** and enter the corresponding data and set the **Disabled for Replication** flag.

**Procedure**

- **Assign Code List to Elements and Systems**
  1. Run transaction **SM30**, enter **MDGV_VM_CODELIST** and choose **Maintain**.
  2. Enter the following data:

<table>
<thead>
<tr>
<th>Type</th>
<th>Global Data Type</th>
<th>Internal List ID</th>
<th>Business System</th>
<th>List ID</th>
<th>List Agency ID</th>
<th>List Version ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTEL</td>
<td>CRMT_SALES_GROUP_MD</td>
<td>50000001 01010 50000002</td>
<td>MDG hub system</td>
<td>VKGRP</td>
<td>CRM client system</td>
<td>1</td>
</tr>
<tr>
<td>DTEL</td>
<td>CRMT_SALES_GROUP_MD</td>
<td>50000001 01010 50000002</td>
<td>MDG hub system</td>
<td>VKGRP</td>
<td>CRM client system</td>
<td>2</td>
</tr>
<tr>
<td>DTEL</td>
<td>CRMT_SALES_GROUP_MD</td>
<td>50000004 01010 50000005</td>
<td>MDG hub system</td>
<td>VKGRP</td>
<td>CRM client system</td>
<td>3</td>
</tr>
</tbody>
</table>
Maintain Value Mapping

1. Run transaction **SM30**, enter **MDGV_ELEMENT** and choose **Display**.

2. Mark **CRMT_SALES_OFFICE_MD** and choose (Display mapping relationships) and then choose (Display -> Change).

1. Under **Assign Code Lists** enter the following data:

<table>
<thead>
<tr>
<th>Table 65</th>
<th>Mapping ID</th>
<th>List Agency ID</th>
<th>List ID</th>
<th>List Version ID</th>
<th>Internal List ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&lt;CRM client system&gt;</td>
<td>VKBUR 1</td>
<td>O 50000001 0101</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>&lt;CRM client system&gt;</td>
<td>VKBUR 2</td>
<td>O 50000004 0101</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Mark **Mapping ID 1**, choose **Define Value Mapping** and enter the data shown below:

<table>
<thead>
<tr>
<th>Table 66</th>
<th>Mapping Comb.</th>
<th>Internal Code Value</th>
<th>External Code Value</th>
<th>Inb. Default</th>
<th>Outb. Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>O 50000002</td>
<td>0001</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>O 50000026</td>
<td>0002</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

3. Mark **Mapping ID 2**, choose **Define Value Mapping** and enter the data shown below:

<table>
<thead>
<tr>
<th>Table 67</th>
<th>Mapping Comb.</th>
<th>Internal Code Value</th>
<th>External Code Value</th>
<th>Inb. Default</th>
<th>Outb. Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>O 50000005</td>
<td>0001</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3. Mark **CRMT_SALES_GROUP_MD** and choose (Display mapping relationships).

1. Under **Assign Code Lists** enter the following data:

<table>
<thead>
<tr>
<th>Table 68</th>
<th>Mapping ID</th>
<th>List Agency ID</th>
<th>List ID</th>
<th>List Version ID</th>
<th>Internal List ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&lt;CRM client system&gt;</td>
<td>VKGRP 1</td>
<td>O 50000001 01010 50000002</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>&lt;CRM client system&gt;</td>
<td>VKGRP 2</td>
<td>O 50000001 01010 50000026</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>&lt;CRM client system&gt;</td>
<td>VKGRP 3</td>
<td>O 50000004 01010 50000005</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Mark **Mapping ID 1**, choose **Define Value Mapping** and enter the data shown below:
### Additional Settings

If you want data changes executed in the CRM client system to be stored in the MDG hub system, set up the systems as described:

#### CRM client system

1. Run the Customizing activity [SAP Customizing Implementation Guide > Cross-Application Components > SAP Business Partner > Data Distribution > Activate Function Modules](#).

2. For the function module CRMMDG_BUPX_OUTBOUND set the flag **Call**. If CRMMDG_BUPX_OUTBOUND is missing in the table, enter the following data:

   ![Table 72](#)

#### MDG hub system

1. Run the Customizing activity [Master Data Governance, Central Governance > General Settings > Value Mapping > Define Technical Settings for Business Systems](#).

2. Mark the CRM client system, choose **Define Bus. Systems, BOs** and select the **BO Types 147** for business partner data and **1405** for business partner relationship data.

3. Mark the BO types separately each, select **Define Bus. Systems, BOs and Communication Channel** and enter the following:
   - Communication Channel: **Replication via Services**
   - Key Harmonization: **Key Mapping**
5.6 Value Mapping

You can use value mapping to map the code values for customizing elements that are represented in the system to the code values of a named external list. The external list can be a global code list or a system-specific code list.

Prerequisites

- You understand the meaning of the code values in the various systems and have agreed mappings with business users.
- You have determined whether to use a global code list or a system-specific code list.
  - We recommend you use a global code list if you are using Enterprise Service Oriented Architecture (ESOA) communications or if target systems support value mapping.
  - You must use a system-specific code list if you are using Application Link Enabling (ALE) communications or if target systems do not support value mapping.

SAP target systems in releases EHP4 and below and non-SAP target systems can only support value mapping with the help of SAP NetWeaver Process Integration (PI) tools or other middleware tools.

Features

When implementing value mapping, you have the following options:

- Use of a Global Code List (see Prerequisites section)
  For more information, see Value Mapping: Use of Global Code Lists [page 145].
- Use of a System-Specific Code List (see Prerequisites section)
  For more information, see Value Mapping: Use of System-Specific Code Lists [page 147].

5.6.1 Value Mapping: Use of Global Code Lists

You use this process to implement value mapping using global code lists. We recommend you use a global code list for inbound and outbound mapping to target systems if you are using enterprise service-oriented architecture (eSOA) communications or if target systems support value mapping.

Process

The following steps describe value mapping for a selected value when you use a global code list. The selected value is the Company form of address.

1. **Outbound Processing on the MDG Hub**
When the MDG hub sends a message that contains the Company form of address, the system converts the internal code defined for the MDG hub (0004) to the code defined for the message (0003). See graphic below.

Figure 19: Use of a Global List: – MDG Hub Configuration

The value mapping configuration information shown in the graphic is summarized in the table below.

Table 73

<table>
<thead>
<tr>
<th>Values in the Message</th>
<th>Internal Values (MDG Hub)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Values come from one of the global code lists for the</td>
<td>• Values come from the AD_TITLE data element. This data element is associated with the</td>
</tr>
<tr>
<td>GDT FormOfAddressCode</td>
<td>AD_TITLE domain.</td>
</tr>
<tr>
<td>• Global Code List</td>
<td>• Codes are 0001 = Ms., 0002 = Mr., and 0003 = Company.</td>
</tr>
<tr>
<td>• AD_TITLE</td>
<td>• Values are mapped to the values described in the Values in the Message column.</td>
</tr>
<tr>
<td>• List ID</td>
<td></td>
</tr>
<tr>
<td>• MDG_GLOBAL FORM_OF_ADDRESS S_CODE_C</td>
<td></td>
</tr>
<tr>
<td>• List Agency ID</td>
<td></td>
</tr>
<tr>
<td>• MDG_GLOBAL</td>
<td></td>
</tr>
<tr>
<td>• List Version ID</td>
<td></td>
</tr>
<tr>
<td>• 1</td>
<td></td>
</tr>
<tr>
<td>• Codes are 0001 = Ms., 0002 = Mr., and 0003 = Company.</td>
<td></td>
</tr>
<tr>
<td>• Values are mapped to the values described in the</td>
<td></td>
</tr>
<tr>
<td>Internal Values (MDG Hub) column.</td>
<td></td>
</tr>
</tbody>
</table>

2. Inbound Processing on the Target System

When the target system receives a message that contains the Company form of address, the system converts the internal code defined for the MDG hub (0003) to the code defined for the message (0003). In this case, the values are the same. See graphic below.

Figure 20: Use of a Global List: – Target System Configuration
The value mapping configuration information shown in the graphic is summarized in the table below.

### Table 74

<table>
<thead>
<tr>
<th>Values in the Message</th>
<th>Internal Values (Target System)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Values come from one of the global code lists for the Global Data Type (GDT) FormOfAddressCode. This code list has a list agency ID of MDGLOBAL and a list version ID of 1.</td>
<td>• Values come from the AD_TITLE data element. This data element is associated with the AD_TITLE domain.</td>
</tr>
<tr>
<td>• Codes are 0001 = Ms., 0002 = Mr., and 0003 = Company.</td>
<td>• Codes are 0002 = Ms., 0001 = Mr., and 0003 = Company.</td>
</tr>
<tr>
<td>• Values are mapped to the values described in the Internal Values (Target System) column.</td>
<td>• Values are mapped to the values described in the Values in the Message column.</td>
</tr>
</tbody>
</table>

### 5.6.2 Value Mapping: Use of System-Specific Code Lists

You use this process to implement value mapping using system-specific code lists. You must use a system-specific code list if you are using Application Link Enabling (ALE) communications or if target systems do not support value mapping.

**Process**

The following steps describe value mapping for a selected value when you use a system-specific code list. The selected value is the **Company** form of address.

1. **Outbound Processing on the MDG Hub**

   When the MDG hub sends a message that contains the **Company** form of address, the system converts the internal code defined for the MDG hub (0004) to the code defined for the message (0003). See graphic below.

![Figure 21: Value Mapping: System-Specific Code Lists](image)

The value mapping configuration information shown in the graphic is summarized in the table below.
Table 75

<table>
<thead>
<tr>
<th>Values in the Message</th>
<th>Internal Values (MDG Hub)</th>
</tr>
</thead>
<tbody>
<tr>
<td>● Values come from one of the global code lists for the Global Data Type (GDT) FormOfAddressCode. This code list has a list agency ID of MDG_GLOBAL and a list version ID of 1.</td>
<td>● Values come from the AD_TITLE data element. This data element is associated with the AD_TITLE domain.</td>
</tr>
<tr>
<td>● Codes are 0002 = Ms., 0001 = Mr., and 0003 = Company.</td>
<td>● Codes are 0001 = Ms., 0002 = Mr., and 0004 = Company.</td>
</tr>
<tr>
<td>● Values are mapped to the values described in the Internal Values (MDG Hub) column.</td>
<td>● Values are mapped to the values described in the Values in the Message column.</td>
</tr>
</tbody>
</table>

2. **Inbound Processing on the Target System**

   When the target system receives a message that contains the Company form of address, the system accepts the values as they are without doing any conversion.

5.7 **Key Mapping**

Customizing for key mapping under [General Settings ➔ Key Mapping](#) is recommended but not mandatory. You can use key mapping instantly, without doing any Customizing.

The use cases for Customizing key mapping are as follows:

- **Changing Default Key Mapping Settings**
- **Extending Key Mapping Settings for Existing Business Objects**
  
  Typically, you extend existing key mapping settings by adding new object IDs.
- **Extending Key Mapping Settings for New Business Objects**

The Customizing activities you use to extend key mapping do not enable key mapping. After you define elements such as object IDs, you must implement them in code. You can refer to the code for existing key mapping entries to implement new key mapping entries.

5.8 **Governance Application Programming Interface**

For greater flexibility you want to be able to develop new UIs that enhance your Master Data Governance applications and are consistent with the existing software. A number of developments in the Master Data Governance Application Framework (MDGAF) allow you greater freedom to build UIs for applications.

- Governance API
- Convenience API
- Application Context API
- Communicator
- Change Request UI Building Block (CRUIBB)

The configuration of components is shown below:
All interactions between applications and MDGAF are now handled by either the Governance API or the Convenience API. It is not possible to use the Convenience API and the Governance API at the same time for the same model. This restriction is introduced to prevent misuse of the both APIs.

**Features**

**Governance API**

The Governance API covers the entire governance process, handling processes that are not UI-related, and background services such as master data load and data replication.

The Governance API is designed to handle multiple change requests simultaneously. At any time, one instance of the Governance API can exist in the system per data model.

The Governance API also provides services to the convenience API. There is less grouping of functions than in the Convenience API so that you can combine a greater range of individual methods to meet the needs of the application. The Governance API also provides services for UI issues, but the applications access these services through the Convenience API, which then calls the Governance API.

The Governance API Class ID is `CL_USMD戈OV_API` (IF_USMD_GOV_API).

**Convenience API**

The Convenience API provides the functionality needed for an application to work with a change request. It can handle one change request for a single data model at a given time. The Convenience API takes over all governance-relevant logic such as managing change request data, handling change requests, and routing change requests to the Governance API. The Convenience API groups together some of the methods of the Governance API ensuring tighter control of the change request-handling capability available to the applications, and simplifying the use of UI services for applications. The application manages only the application data.

The Convenience API Class ID is `CL_USMD.Conv_SOM.GOV_API` (IF_USMD_CONV_SOM_GOV_API).
Application Context API

The Application Context API stores context-specific runtime information at a central point so that this information is accessible for other parts of the application and can be used to control the program-flow. Previously the system did not provide application context information such as what change request is being processed and whether the master data object is to be created or updated. The Application Context API provides a consistent, reliable solution to this problem.

The following context information is available:

- Data model
- Business activity
- Workflow information
  - Change request
  - Change request type
  - Change request step
  - Change request index (relevant for parallel processing)
  - Workflow item
- Application parameter data (stored in the Workflow Container, not accessed by MDG)

The Application Context API offers the following advantages:

- Allows existing UIs to access the application context without using the complete Governance API
- Keeps existing interfaces stable
- Increases flexibility.
  - While, for example, the Governance API or Convenience API can only be instantiated for a data model, the Context API is directly available to MDGAF components such as a UI application or background process.
- Manages application-specific context data
  - Application-specific context data is stored within the workflow container. This enables you to provide application-specific context data throughout the workflow.

The Application Context API Class ID is CL_USMD_APP_CONTEXT.

Communicator

The Communicator allows the user to work with the change request and ensures consistency of change request handling prerequisites, such as change request type, change request ID, and work item ID. When a user begins working with a change request, the Communicator recognizes missing parameters and initiates user interaction accordingly, for example, requesting the user to specify a change request type if none has yet been specified.

Change Request UI Building Block (CRUIBB)

This UI component is included in application-specific UIs and handles the presentation of change request data in Web Dynpro applications, ensuring a consistent UI layout for change request data across all applications. The CRUIBB contains data such as CR description, priority, reason for CR, notes, and attachments. Applications need to manage the application data only.
6 Time Dependency

MDG-C supports time dependency and time validity in restricted scenarios. In the MDG context time dependency means that one date field is a key field and you can maintain multiple intervals per record. Time validity means that both date fields are only attributes and you can maintain one validity interval per record.

Features

Time dependency for addresses (BUT020) and address usages (BUT021_FS)
MDG does not support time dependency for addresses and address usages. Nevertheless, time dependency can be switched on in MDG systems without any damage to data. MDG uses the data of the record that is currently valid.

Time dependency for BP relationships (BUT050)
MDG supports time-dependent business partner relationships. For more information about time dependency for BP relationships, see 2249877.

Time validity for bank accounts (BUT0BK), BP ID numbers (BUT0ID), and BP roles (BUT100)
MDG supports time validity. You can add fields from the field list using the screen configuration.

Constraints

Time dependency for central BP data (BUT000)

⚠️ Caution
MDG does not support time dependency for central business partner data. The switch is not allowed to be active within an MDG system.
7 Interlocking

Interlocking specifies which nodes are interlocked with a pending change request while a change to a hierarchy is made. A change to a hierarchy can comprise adding a node, moving a node, removing a node, changing the attributes of a node, or creating a hierarchy. After a change to a hierarchy is saved to a change request, changes to interlocked nodes must be saved to the same change request. If a node is not interlocked, you can use any change request to make a hierarchy-specific change.

With a setting of *Loose*, nodes assigned to the parent node of the node being changed are interlocked.

With a setting of *Strict*, interlocking propagates upwards and downwards from the parent node of the node being changed as follows:

- **Upwards interlocking** interlocks the parent node and its assigned nodes, the parent node of the parent node and its assigned nodes, and so on up to the root node.
- **Downwards interlocking** interlocks child nodes of the parent node, their child nodes, and so on down to the end nodes. This comprises a subhierarchy of interlocked nodes with the parent node at its root.

For a full description of what interlocking means that includes a graphical representation of the *Loose* and *Strict* settings, see *Scope for Hierarchy-Specific Changes* [page 111].
# Typographic Conventions

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;Example&gt;</code></td>
<td>Angle brackets indicate that you replace these words or characters with appropriate entries to make entries in the system, for example, “Enter your <code>&lt;User Name&gt;</code>.”</td>
</tr>
<tr>
<td>Example ➔ Example</td>
<td>Arrows separating the parts of a navigation path, for example, menu options</td>
</tr>
<tr>
<td>Example</td>
<td>Emphasized words or expressions</td>
</tr>
<tr>
<td><a href="http://www.sap.com">www.sap.com</a></td>
<td>Textual cross-references to an internet address</td>
</tr>
<tr>
<td><code>/example</code></td>
<td>Quicklinks added to the internet address of a homepage to enable quick access to specific content on the Web</td>
</tr>
<tr>
<td>123456</td>
<td>Hyperlink to an SAP Note, for example, SAP Note <a href="#">123456</a></td>
</tr>
</tbody>
</table>
| Example | • Words or characters quoted from the screen. These include field labels, screen titles, pushbutton labels, menu names, and menu options.  
• Cross-references to other documentation or published works |
| Example | • Output on the screen following a user action, for example, messages  
• Source code or syntax quoted directly from a program  
• File and directory names and their paths, names of variables and parameters, and names of installation, upgrade, and database tools |
| EXAMPLE | Technical names of system objects. These include report names, program names, transaction codes, database table names, and key concepts of a programming language when they are surrounded by body text, for example, `SELECT` and `INCLUDE` |
| EXAMPLE | Keys on the keyboard |