



PUBLIC

How-To: Extend MDG-S / MDG-C by a new custom-defined field (reuse option)

Applicable Releases:
From EhP6 for SAP ERP 6.0 and from S/4HANA 1511

Version 4.0
December 2024

Document History

Document Version	Description
1.0	First official release of this guide (March 2012)
2.0	Layout (July 2020)
3.0	Additional information (August 2022)
4.0	Additional information (December 2024)

1. BUSINESS SCENARIO	4
1.1. DATA MODEL	4
2. STEP BY STEP EXPLANATION	5
2.1. EXTENDED FIELD EXISTS IN BUSINESS PARTNER ONLY	5
2.1.1. Extend the BUT000 Data Model	7
2.1.2. Extend the MDG Data Model	9
2.1.3. Generate Data Model-Specific Structures for MDG	11
2.1.4. Define SMT Mapping	13
2.1.5. Check Customizing	19
2.1.6. Define Field Properties for UI	20
2.1.7. Custom Handler Class	20
2.1.8. Define Field Mapping for SAP HANA-Based Search	21
2.1.9. Enhance Search UIBB	23
2.1.10. Adjust Staging Area of Linked Change Requests	25
2.1.11. Test Your Configuration	25
2.2. EXTENDED FIELD EXISTS IN VENDOR / CUSTOMER ONLY	27
2.2.1. Extend the MDG Data Model	28
2.2.2. Generate Data Model-Specific Structures for MDG	31
2.2.3. Define SMT Mapping	33
2.2.4. Test Your Configuration	39
2.3. EXTENDED FIELD EXISTS IN BUSINESS PARTNER AND IN VENDOR / CUSTOMER	41
2.3.1. Extend LFA1	42
2.3.2. Implement Enhancement Spot CVI_CUSTOM_MAPPER	45
2.3.3. Test Your Configuration	47
2.4. USER INTERFACE CONFIGURATION	48
2.4.1. Adaptation Options in Floorplan Manager	48
2.4.2. Adding a New Field to UI for Single Maintenance (BUT000 Extension Case)	49
2.4.3. Adding a New Field to the UI for Single Maintenance (Extended Field Exists in Vendor / Customer Only)	
	51
3. ADDITIONAL INFORMATION	54
3.1. FURTHER READING	54
3.1.1. Information on SAP MDG on SAP S/4HANA	54
3.1.2. SAP Roadmap Explorer	54
3.1.3. Related Information	54
3.2. SAP NOTES	54
4. APPENDIX	56
4.1. EXAMPLE CODING FOR CLASS ZCL_CUSTOM_BP_HANDLER	56

1. BUSINESS SCENARIO

SAP Master Data Governance provides business processes to find, create, change, and mark master data for deletion. It supports the governance of master data in a central hub and the distribution to connected operational and business intelligence systems.

Domain-specific content (data models, user interfaces, workflows) is provided as part of the standard for several application areas. It is a common requirement from customers to adapt the MDG data models to their specific needs.

This How To Guide describes how to extend the data model BP by adding attributes that already exist as database fields in the appropriate customer includes.

1.1. Data Model

In SAP Master Data Governance data can be stored in the staging area or in the active area. The staging area temporarily stores the working version of master data that is undergoing change within a change request process. The active area permanently stores active instances of master data after changes have been approved by a change request process.

Extending the data model in SAP Master Data Governance with custom attributes involves changes to the data structures in the active area and in the staging area. It also involves defining a mapping between these.

You can extend the data model BP by adding attributes that already exist as database fields in the appropriate ERP tables. This method of extending the data model is called the MDG reuse option.

Your implementation of the reuse option depends on the ERP tables that store the master data after it is activated. The following options are available:

- Extended field exists in Business Partner
- Extended field exists in Vendor / Customer

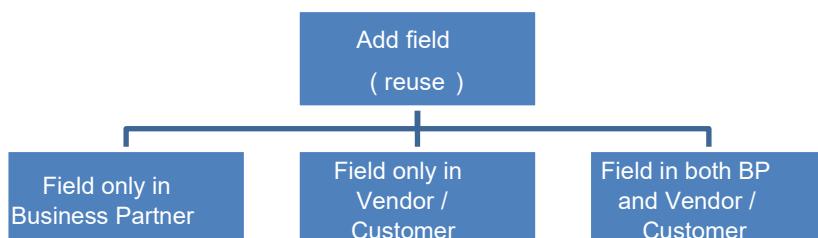
The new field is only extended as a database field in the customer include *Ext. Interface: Central Vendor Data / Data Fields (VMDS_EI_VMD_CENTRAL_DATA)* for the vendor data model as well as in either the customer include *Ext. Interface: Central Vendor Data / X-Fields (VMDS_EI_VMD_CENTRAL_DATA_XFLAG)* for the vendor data model or in the equivalent customer include for the customer data model.

- Extended Field Exists in Business Partner and in Vendor / Customer

The new field is extended as a database field in both the Business Partner and in customer includes for vendor / customer.

Each of the sections in this document contains detailed step-by-step implementation instructions. Follow the steps in the section that is appropriate for your scenario.

These cases are also shown in the figure below.



Note:

You can create new attributes in the customer namespace, for example, under ZZ* and YY*.

2. STEP BY STEP EXPLANATION

Even though we have to distinguish three cases for the field extension, the implementation steps are very similar. The implementation involves the following main tasks:

- Extending the ERP data model
- Extending the MDG data model
- Generating the MDG data model specific structures
- Defining SMT mapping
- Implement CVI mapping (if applicable)

The main differences in the implementation approaches can be identified from the table below. During the step “define SMT mapping” it is important to map to the right structures. If the field extension was made in both business partner and vendor / customer you must additionally implement the CVI mapping (but only SMT mapping to BP is necessary).

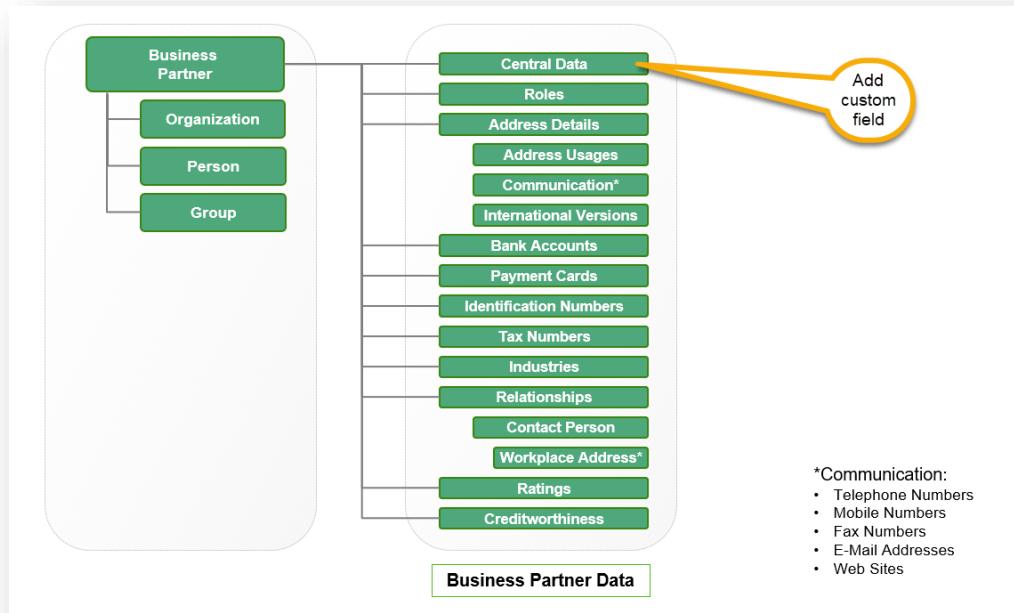
Implementation Steps	Field only in Business Partner	Field only in Vendor/Customer	Field in both BP and Vendor/ Customer
Extend ERP data model	✓	✓	✓
Extend MDG data model	✓	✓	✓
Generate MDG data model	✓	✓	✓
Define SMT mapping	✓ ¹	✓ ²	✓ ¹
Implement CVI mapping			✓

Table: Implementation steps for extending the MDG-S/C data model by a new field

SMT Mappings
¹ MDG staging ↔ Business Partner
² MDG staging ↔ Vendor / Customer

2.1. Extended Field Exists in Business Partner Only

You want to extend the MDG data model for Business Partner by additional fields. The fields are added to the Central Data entity type as shown in the figure below.



In this section, you learn how to add a new attributes **ZZLOB** and **ZZSTAGE** to the MDG **BP** data model. You also must have extended the **BUT000** table on the MDG hub and client systems.

Field Short Description	Scenario	Backend BUT000		MDG BP on BP_CENTRL	
		Field	Data Element	Attribute	Data Element
Line of Business	same field name and same data element	ZZLOB	ZZ9_BUT000_LOB	ZZLOB	ZZ9_BUT000_LOB
Ext. BP Stage	different field name, but same data element	ZZ1_ZEXT_WKF1_BUS	ZZ1_ZEXT_WKF1	ZZSTAGE	ZZ1_ZEXT_WKF1

Note:

Make sure that at least the data elements in the backend and MDG are identical.

This section is subdivided in to one sub-section for each implementation step. Take a look at the table below to get an overview of the steps involved.

In order to facilitate data transfer between the MDG staging area and the **BUT000** table on the MDG hub you have to extend the SMT mapping.

Implementation Steps	Field only in Business Partner	Field only in Vendor/Customer	Field in both BP and Vendor/ Customer
Extend ERP data model	✓	✓	✓
Extend MDG data model	✓	✓	✓
Generate MDG data model	✓	✓	✓
Define SMT mapping	✓ ¹	✓ ²	✓ ¹
Implement CVI mapping			✓

SMT Mappings

1

MDG staging ↔ Business Partner
² MDG staging ↔ Vendor / Customer

2.1.1. Extend the BUT000 Data Model

Extending a backend table with a custom field is not described in this guide. References on how this can be achieved are listed below.

Custom fields can be created not only using the SAP Fiori app Custom Fields and Logic but also using back-end transactions to create database fields.

Custom Fields App and Custom Logic App:

More Information can be found in the SAP Online Help and blogs.

Documentation	Title	Link
SAP Online Help	Custom Fields App and Custom Logic App	https://help.sap.com/docs/ABAP_PLATFORM_NEW/ed0e11412f9841e7ac5cd9a6799368d4/57909455bf7c4fd8bcf48d76c1eae33.html?version=latest
SAP Online Help	App Extensibility: App Extensibility: Manage Customer Master Data	https://help.sap.com/docs/SAP_S4HANA_ON-PREMISE/74b0b157c81944ffaac6ebc07245b9dc/5f328757efd43925e10000000a4450e5.html?version=latest
SAP Online Help	App Extensibility: Manage Supplier Master Data	https://help.sap.com/docs/SAP_S4HANA_ON-PREMISE/74b0b157c81944ffaac6ebc07245b9dc/862f8757efd43925e10000000a4450e5.html?version=latest

Back-end transactions:

More Information can be found in the SAP Online Help and blogs. Consider using existing extension includes.

Documentation	Title	Link
SAP Online Help	Enabling Custom Database Fields for Usage in SAP Fiori Applications	https://help.sap.com/docs/ABAP_PLATFORM_NEW/ed0e11412f9841e7ac5cd9a6799368d4/4accfedc4d2e49c1b321b6ebf288a430.html?version=latest
SAP Online Help	SAP Business Partner Extensibility	https://help.sap.com/docs/BS_CA/725b6ad626684262b8d74a55fc8ed089/a6af8d5377a0ec23e10000000a174cb4.html?version=latest
Blog	SAP S/4HANA Cookbook Customer/Vendor Integration	https://support.sap.com/content/dam/SAAP/SAP_ActivateS4H.0781%20SAP%20S4HANA%20Cookbook%20Customer%20Vendor%20Integration.pdf

You can extend the tables listed below using the corresponding customer include.

Table	Description	Customer Include	Customer Include
BUT000	General	INCL_EEW_BUT000	INCL_EEW_BUT000_X
KNA1	Customer General	INCL_EEW_KNA1	INCL_EEW_KNA1_X
KNB1	Company Code (Customer)	INCL_EEW_KNB1	INCL_EEW_KNB1_X
KNVV	Sales Data	INCL_EEW_KNVV	INCL_EEW_KNVV_X
LFA1	Supplier General	INCL_EEW_LFA1	INCL_EEW_LFA1_X
LFB1	Company Code (Supplier)	INCL_EEW_LFB1	INCL_EEW_LFB1_X
LFM1	Purchasing Org Data	INCL_EEW_LFM1	INCL_EEW_LFM1_X

The screenshots below show the extension with custom field **ZZLOB** and **ZZ1_ZEXT_WKF1_BUS** (field was created using the Key User Extensibility) for general data of business partner (**BUT000** and **structure BUS_EI_BUPA_CENTRAL_DATA_XFLAG**). This concerns the structure **INCL_EEW_BUT000** and **INCL_EEW_BUT000_X**.

SAP Dictionary: Display Table

Transparent Table: BUT000 Active
Short Description: BP: General data I

Attributes Delivery and Maintenance Fields Entry help/check Currency/Quantity Fields Indexes

1 / 148

Field	Key	Initial... Data element	Data Type	Length	Decimal...	KoordSystem	Short Description	Group
<input type="checkbox"/> CLIENT	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0		0 Client	
<input type="checkbox"/> PARTNER	<input checked="" type="checkbox"/>	BU_PARTNER	CHAR	10	0		0 Business Partner Number	
<input type="checkbox"/> ..INCLUDE	<input type="checkbox"/>	BUS000_DAT	STRU	0	0		0 BP: General Data I (Data Fields - External)	
<input type="checkbox"/> ..INCLUDE	<input type="checkbox"/>	BUS000A	STRU	0	0		0 CBP: General data (independent of partner cat.)	INDEP
<input type="checkbox"/> ..INCLUDE	<input type="checkbox"/>	BUS000AINI	STRU	0	0		0 CBP: General Data (not dep. on Part. Cat.), Initial Screen	
<input type="checkbox"/> ..TYPE	<input type="checkbox"/>	BU_TYPE	CHAR	1	0		0 Business Partner Category	
<input type="checkbox"/> BPKIND	<input type="checkbox"/>	BU_BPKIND	CHAR	4	0		0 Business Partner Type	
<input type="checkbox"/> BU_GROUP	<input type="checkbox"/>	BU_GROUP	CHAR	4	0		0 Business Partner Grouping	
<input type="checkbox"/> BPEXT	<input type="checkbox"/>	BU_BPEXT	CHAR	20	0		0 Business Partner Number in External System	
<input type="checkbox"/> ..INCLUDE	<input type="checkbox"/>	BUS000ADAT	STRU	0	0		0 CBP: General Data (not dep. on Part. Cat.), Data Screens	
<input type="checkbox"/> BU_SORT1	<input type="checkbox"/>	BU_SORT1	CHAR	20	0		0 Search term 1 for business partner	
<input type="checkbox"/> BU_SORT2	<input type="checkbox"/>	BU_SORT2	CHAR	20	0		0 Search term 2 for business partner	
<input type="checkbox"/> SOURCE	<input type="checkbox"/>	BU_SOURCE	CHAR	4	0		0 Data Origin Types	
<input type="checkbox"/> TITLE	<input type="checkbox"/>	AD_TITLE	CHAR	4	0		0 Form-of-Address Key	
<input type="checkbox"/> XDELETE	<input type="checkbox"/>	BU_XDELETE	CHAR	1	0		0 Central Archiving Flag	
<input type="checkbox"/> XBLCK	<input type="checkbox"/>	BU_XBLCK	CHAR	1	0		0 Central Block for Business Partner	
<input type="checkbox"/> AUGRP	<input type="checkbox"/>	BU_AUGRP	CHAR	4	0		0 Authorization Group	
<input type="checkbox"/> TITLE_LET	<input type="checkbox"/>	BU_TITLE_LET	CHAR	50	0		0 Salutation	
<input type="checkbox"/> BU_LOGSYS	<input type="checkbox"/>	LOGSYS	CHAR	10	0		0 Logical system	
<input type="checkbox"/> CONTACT	<input type="checkbox"/>	BU_CONTACT	CHAR	1	0		0 Business Partner: Contact Permission	
<input type="checkbox"/> NOT_RELEASED	<input type="checkbox"/>	BU_XNOT_RELEASED	CHAR	1	0		0 Indicator: Not Released	
<input type="checkbox"/> NOT_LG_COMPETENT	<input type="checkbox"/>	BU_NOT_LEG_COMPETE	CHAR	1	0		0 Indicator: Not Legally Competent	
<input type="checkbox"/> PRINT_MODE	<input type="checkbox"/>	BU_PRINT_MODE	CHAR	1	0		0 Business Partner Print Format	
<input type="checkbox"/> ..INCLUDE	<input type="checkbox"/>	INCL_EEW_BUT000	STRU	0	0		0 EEW Extension of BUT000 (Contains CI)	INCL_EEW_BUT000

Structure: INCL_EEW_BUT000 Active
Short Description: EEW Extension of BUT000 (Contains CI)

Attributes Components Input Help/Check Currency/quantity fields

25 / 63

Component	Typing Method	Component Type	Data Type	Length	Decimal...	KoordSystem	Short Description
<input type="checkbox"/> ZZ1_TELEPHONE_BUS	1 Types	ZZ1_TELEPHONE	CHAR	40	0		0 Telephone
<input type="checkbox"/> APPEND	1 Types	ZZ1_GOT2AOJCZ2CI4U	---	0	0		0 ZZ1_ZEXT_WKF1
<input checked="" type="checkbox"/> ZZ1_ZEXT_WKF1_BUS	1 Types	ZZ1_ZEXT_WKF1	CHAR	3	0		0 Ext. BP Stage
<input type="checkbox"/> APPEND	1 Types	ZZ1_HOT4KR4QNAHE_IK	---	0	0		0 ZZ1_ZEXT_AHBPGeneralNum

Structure: INCL_EEW_BUT000 Active
Short Description: EEW Extension of BUT000 (Contains CI)

Attributes Components Input Help/Check Currency/quantity fields

61 / 63

Component	Typing Method	Component Type	Data Type	Length	Decimal...	KoordSystem	Short Description
<input type="checkbox"/> ZZ1_BPSIZE_BUS	1 Types	ZZ1_BPSIZE	CHAR	3	0		0 BPSIZE
<input type="checkbox"/> APPEND	1 Types	ZZ9_LOB	---	0	0		0 Extensibility field Line of Business for MDG
<input type="checkbox"/> ZZLOB	1 Types	ZZ9_BUT000_LOB	CHAR	4	0		0 Extensibility data element for Line of Business

Structure: **BUS_EI_BUPA_CENTRAL_DATA_XFLAG** Active

Short Description: External Interface: Indicator for Creating a Partner

Attributes **Components** Input Help/Check Currency/quantity fields

Predefined Type

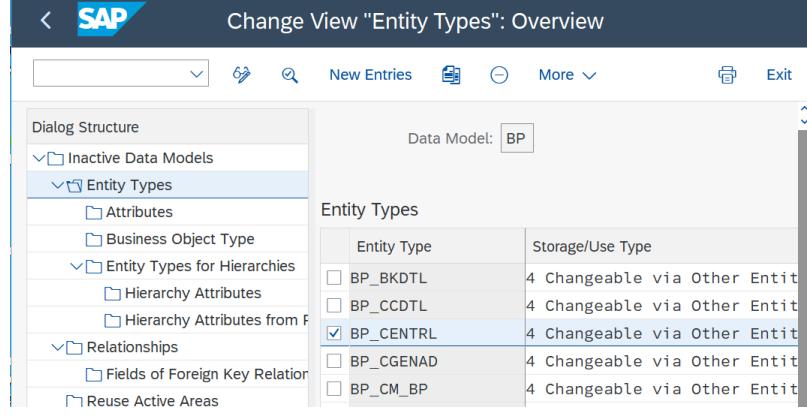
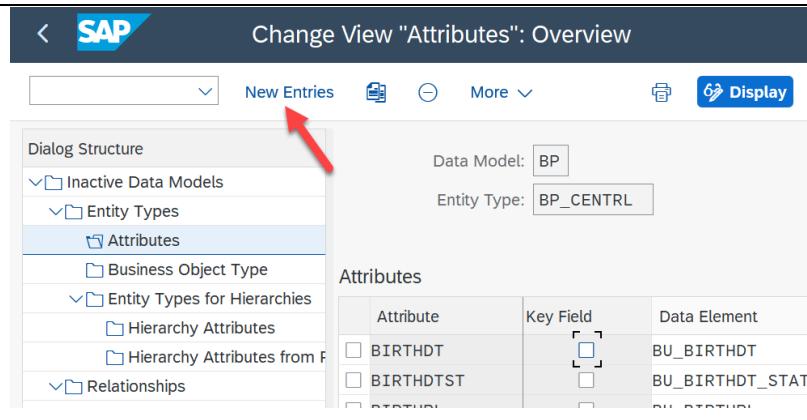
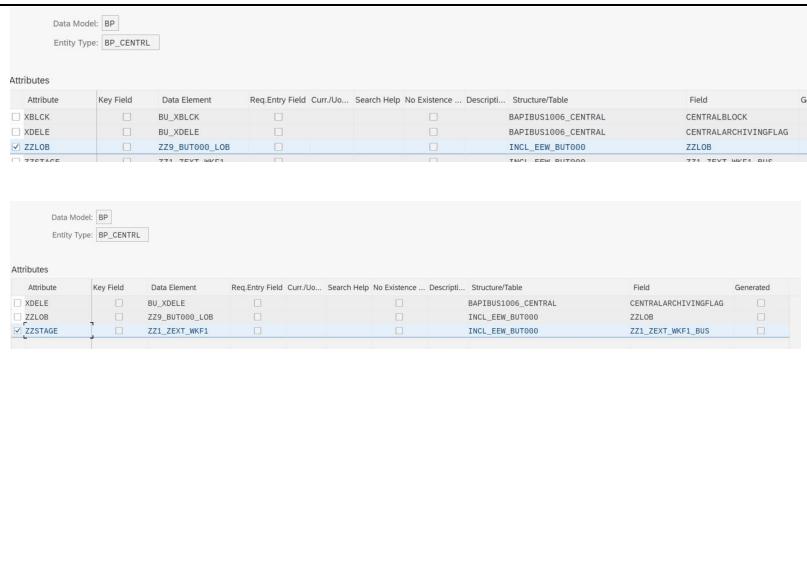
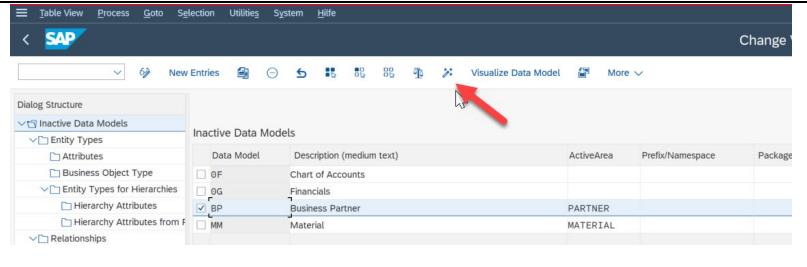
Component	Typing Method	Component Type	Data Type	Length	Decimal...	KoordSystem	Short Description
BP_CENTRALDATA	1 Types	BUS_EI_STRUC_CENTR	...	0	0		0 Ext. Interface: Structure CENTRAL_X
BP_PERSON	1 Types	BUS_EI_STRUC_CENTR	...	0	0		0 Ext. Interface: Structure CENTRAL_PERSON_X
BP_ORGANIZATION	1 Types	BUS_EI_STRUC_CENTR	...	0	0		0 Ext. Interface: Structure CENTRAL_ORGANIZATION_X
BP_GROUP	1 Types	BUS_EI_STRUC_CENTR	...	0	0		0 Ext. Interface: Structure CENTRAL_GROUP_X
INCLUDE	1 Types	INCL_EEW_BUT000_X	...	0	0		0 Checkbox Structure of EEW Extension of BUT000(Contains CI_X)

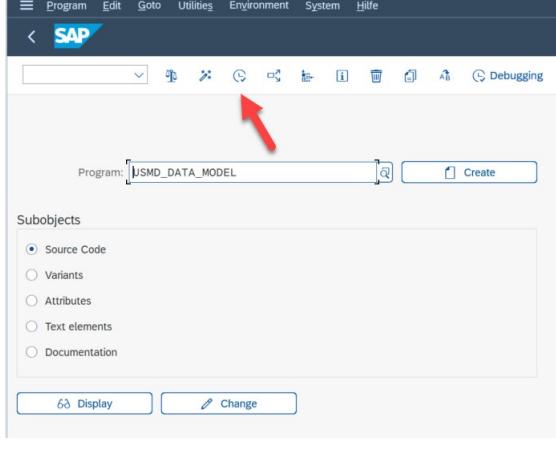
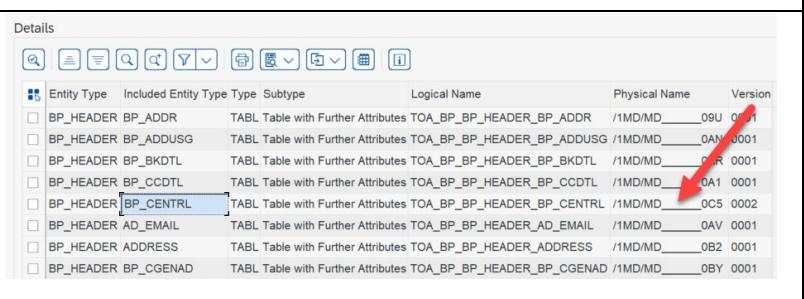
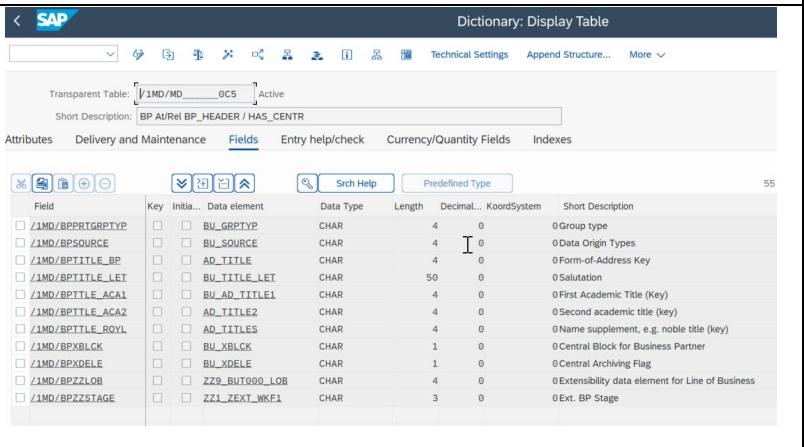
Consider enabling existing custom database fields for usage in SAP Fiori applications

(https://help.sap.com/docs/SAP_NETWEAVER_AS_ABAP_752/b4367b1cec3243c4989f0ff3d727c4ab/4accfecd4d2e49c1b321b6ebf288a430.html?version=latest) and run transaction **SCFD_EUI**.

2.1.2. Extend the MDG Data Model

1	Log into system for cross-client maintenances.	
2	<p>Start Customizing for <i>Master Data Governance</i> (transaction MDGIMG).</p> <p>Choose General Settings > Data Modeling > Edit Data Model.</p>	
3	Select the <i>BP</i> data model.	

4	<p>Choose <i>Entity Types</i>.</p> <p>Select <i>Entity Types BPBP_CENTRL</i>.</p>  <table border="1"> <thead> <tr> <th>Entity Type</th> <th>Storage/Use Type</th> </tr> </thead> <tbody> <tr> <td>BP_BKDTL</td> <td>4 Changeable via Other Entit</td> </tr> <tr> <td>BP_CCDTL</td> <td>4 Changeable via Other Entit</td> </tr> <tr> <td>BP_CENTRL</td> <td>4 Changeable via Other Entit</td> </tr> <tr> <td>BP_GENAD</td> <td>4 Changeable via Other Entit</td> </tr> <tr> <td>BP_CM_BP</td> <td>4 Changeable via Other Entit</td> </tr> </tbody> </table>	Entity Type	Storage/Use Type	BP_BKDTL	4 Changeable via Other Entit	BP_CCDTL	4 Changeable via Other Entit	BP_CENTRL	4 Changeable via Other Entit	BP_GENAD	4 Changeable via Other Entit	BP_CM_BP	4 Changeable via Other Entit																																											
Entity Type	Storage/Use Type																																																							
BP_BKDTL	4 Changeable via Other Entit																																																							
BP_CCDTL	4 Changeable via Other Entit																																																							
BP_CENTRL	4 Changeable via Other Entit																																																							
BP_GENAD	4 Changeable via Other Entit																																																							
BP_CM_BP	4 Changeable via Other Entit																																																							
5	<p>Choose <i>Attributes</i>.</p> <p>Choose the <i>New Entries</i> pushbutton to add a new attribute to the <i>BP_CENTRL</i> entity type.</p>  <table border="1"> <thead> <tr> <th>Attribute</th> <th>Key Field</th> <th>Data Element</th> </tr> </thead> <tbody> <tr> <td>BIRTHDHT</td> <td><input checked="" type="checkbox"/></td> <td>BU_BIRTHDHT</td> </tr> <tr> <td>BIRTHDHTST</td> <td><input checked="" type="checkbox"/></td> <td>BU_BIRTHDHT_STAT</td> </tr> <tr> <td>ZZSTAGE</td> <td><input checked="" type="checkbox"/></td> <td>INCL_EEW_BUT000</td> </tr> </tbody> </table>	Attribute	Key Field	Data Element	BIRTHDHT	<input checked="" type="checkbox"/>	BU_BIRTHDHT	BIRTHDHTST	<input checked="" type="checkbox"/>	BU_BIRTHDHT_STAT	ZZSTAGE	<input checked="" type="checkbox"/>	INCL_EEW_BUT000																																											
Attribute	Key Field	Data Element																																																						
BIRTHDHT	<input checked="" type="checkbox"/>	BU_BIRTHDHT																																																						
BIRTHDHTST	<input checked="" type="checkbox"/>	BU_BIRTHDHT_STAT																																																						
ZZSTAGE	<input checked="" type="checkbox"/>	INCL_EEW_BUT000																																																						
6	<p>Enter the attribute name ZZLOB, the same data element used in the backend table, the used structure and field name. Enter the attribute name ZZSTAGE (differs from backend), but the same data element used in the backend table, the used structure and field name.</p> <p>Note: Make sure that at least the data element in the backend and MDG is identical.</p>  <table border="1"> <thead> <tr> <th>Attribute</th> <th>Key Field</th> <th>Data Element</th> <th>Req.Entry Field</th> <th>Curr.Uo...</th> <th>Search Help</th> <th>No Existence...</th> <th>Description...</th> <th>Structure/Table</th> <th>Field</th> <th>Generated</th> </tr> </thead> <tbody> <tr> <td>XBLOCK</td> <td><input type="checkbox"/></td> <td>BU_XBLOCK</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>BAPIBUS1006_CENTRAL</td> <td>CENTRALBLOCK</td> <td><input type="checkbox"/></td> </tr> <tr> <td>XDELE</td> <td><input type="checkbox"/></td> <td>BU_XDELE</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>BAPIBUS1006_CENTRAL</td> <td>CENTRALARCHIVINGFLAG</td> <td><input type="checkbox"/></td> </tr> <tr> <td>ZZLOB</td> <td><input checked="" type="checkbox"/></td> <td>Z29_BUT000_LOB</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>INCL_EEW_BUT000</td> <td>ZZLOB</td> <td><input type="checkbox"/></td> </tr> <tr> <td>ZZSTAGE</td> <td><input checked="" type="checkbox"/></td> <td>ZZ1_ZEXT_WKF1</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>INCL_EEW_BUT000</td> <td>ZZ1_ZEXT_WKF1_BUS</td> <td><input type="checkbox"/></td> </tr> </tbody> </table>	Attribute	Key Field	Data Element	Req.Entry Field	Curr.Uo...	Search Help	No Existence...	Description...	Structure/Table	Field	Generated	XBLOCK	<input type="checkbox"/>	BU_XBLOCK	<input type="checkbox"/>	BAPIBUS1006_CENTRAL	CENTRALBLOCK	<input type="checkbox"/>	XDELE	<input type="checkbox"/>	BU_XDELE	<input type="checkbox"/>	BAPIBUS1006_CENTRAL	CENTRALARCHIVINGFLAG	<input type="checkbox"/>	ZZLOB	<input checked="" type="checkbox"/>	Z29_BUT000_LOB	<input type="checkbox"/>	INCL_EEW_BUT000	ZZLOB	<input type="checkbox"/>	ZZSTAGE	<input checked="" type="checkbox"/>	ZZ1_ZEXT_WKF1	<input type="checkbox"/>	INCL_EEW_BUT000	ZZ1_ZEXT_WKF1_BUS	<input type="checkbox"/>																
Attribute	Key Field	Data Element	Req.Entry Field	Curr.Uo...	Search Help	No Existence...	Description...	Structure/Table	Field	Generated																																														
XBLOCK	<input type="checkbox"/>	BU_XBLOCK	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BAPIBUS1006_CENTRAL	CENTRALBLOCK	<input type="checkbox"/>																																														
XDELE	<input type="checkbox"/>	BU_XDELE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BAPIBUS1006_CENTRAL	CENTRALARCHIVINGFLAG	<input type="checkbox"/>																																														
ZZLOB	<input checked="" type="checkbox"/>	Z29_BUT000_LOB	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	INCL_EEW_BUT000	ZZLOB	<input type="checkbox"/>																																														
ZZSTAGE	<input checked="" type="checkbox"/>	ZZ1_ZEXT_WKF1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	INCL_EEW_BUT000	ZZ1_ZEXT_WKF1_BUS	<input type="checkbox"/>																																														
7	<p>Save and activate the changes.</p>  <table border="1"> <thead> <tr> <th>Data Model</th> <th>Description (medium text)</th> <th>ActiveArea</th> <th>Prefix/Namespace</th> <th>Package</th> </tr> </thead> <tbody> <tr> <td>0F</td> <td>Chart of Accounts</td> <td></td> <td></td> <td></td> </tr> <tr> <td>0G</td> <td>Financials</td> <td></td> <td></td> <td></td> </tr> <tr> <td>BP</td> <td>Business Partner</td> <td>PARTNER</td> <td></td> <td></td> </tr> <tr> <td>MM</td> <td>Material</td> <td>MATERIAL</td> <td></td> <td></td> </tr> </tbody> </table>	Data Model	Description (medium text)	ActiveArea	Prefix/Namespace	Package	0F	Chart of Accounts				0G	Financials				BP	Business Partner	PARTNER			MM	Material	MATERIAL																																
Data Model	Description (medium text)	ActiveArea	Prefix/Namespace	Package																																																				
0F	Chart of Accounts																																																							
0G	Financials																																																							
BP	Business Partner	PARTNER																																																						
MM	Material	MATERIAL																																																						

8	<p>Start transaction SE38. Run program USMD_DATA_MODEL and choose Data Model: BP</p>	
9	<p>Locate entity type BP_CENTRL and choose it to see the generated table.</p>	
10	<p>Verify that the generated table contains the added field /1MD/BPZZLOB and /1MD/BPZZSTAGE.</p>	

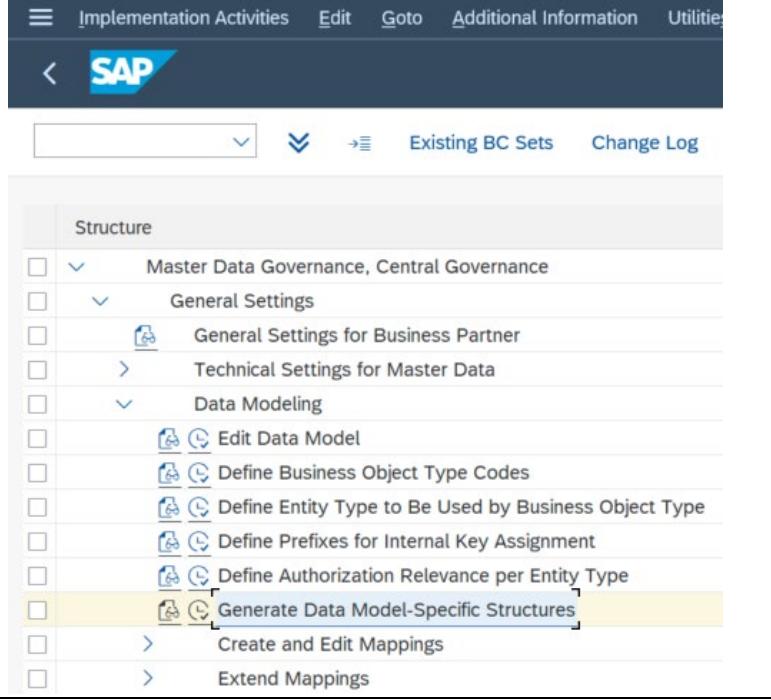
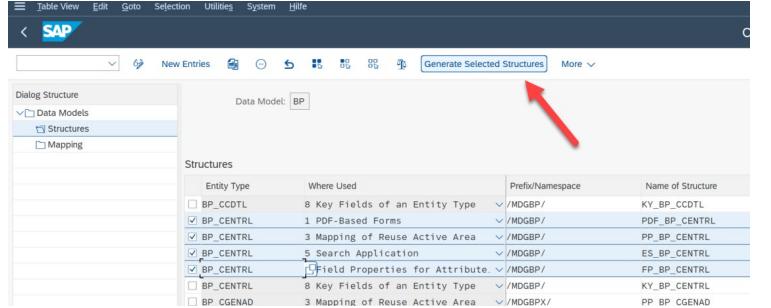
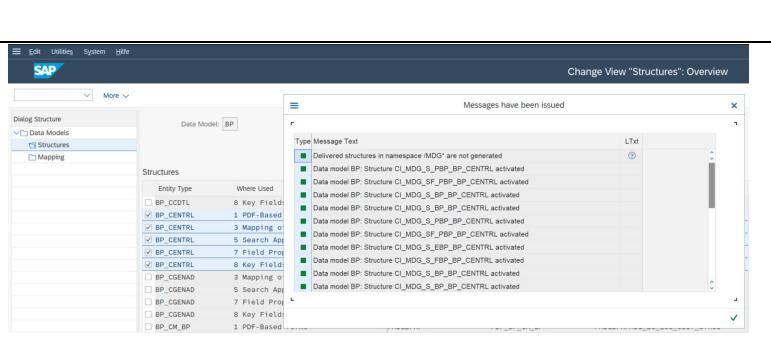
2.1.3. Generate Data Model-Specific Structures for MDG

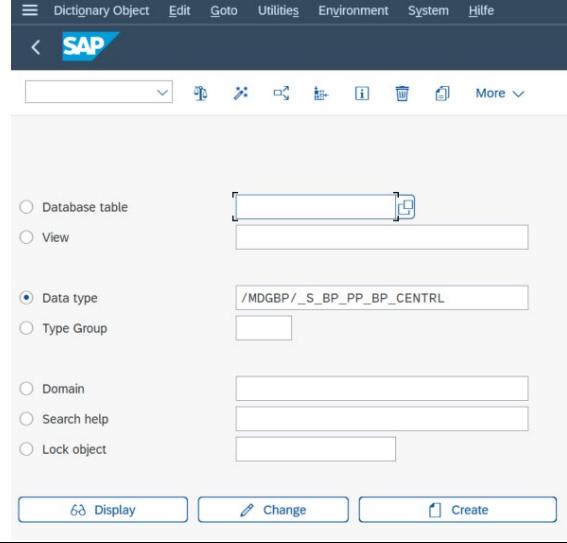
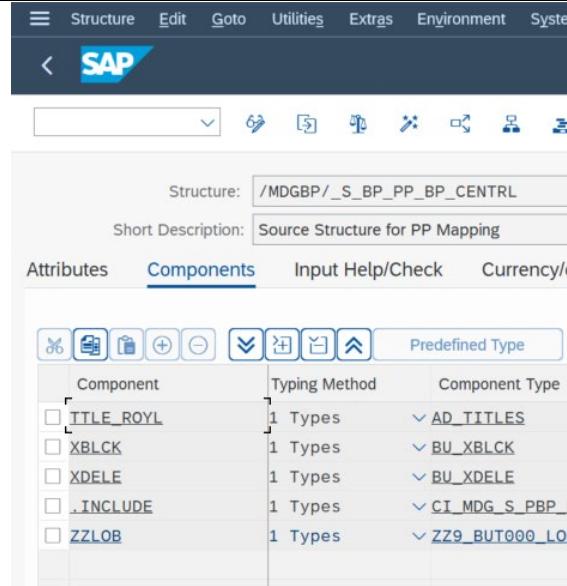
In this Customizing activity, you generate the following technical structures in the ABAP Dictionary for each data model and entity type. If you have generated the data model specific structures using the [data model activation](#) of the Customizing activity [Edit Data Model](#), then you do not need to use the Customizing activity [Generate Data Model-Specific Structures](#) for the generation of data model-specific structures. SAP delivers both data models and corresponding structures. The namespace of these structures is /MDG*. These structures also contain a [customizing include](#). If you change the data model later, you need to adjust the structures by means of these customizing includes.

Note:

In general if you change a data model (for example, if you change attributes of entity types or relationships); you need to regenerate the structures.

1	<p>Log into the system for cross-client maintenance.</p>	
---	----------------------------------------------------------	--

2	<p>Start Customizing for <i>Master Data Governance</i> (transaction MDGIMG).</p> <p>Choose <i>General Settings > Data Modeling > Generate Data Model-Specific Structures</i>.</p>	
3	<p>Select the row with <i>BP Data Model</i>. Remember the development package.</p> <p>Choose <i>Structures</i> in the left-hand panel. Select the structures of the corresponding entity types.</p> <p>Choose the <i>Generate Structures</i> pushbutton.</p>	
4	<p>Assign development package /MDGBP/MDG_BS_BP_STRUCTURES, if a dialog box appears (several times).</p> <p>The structures in namespace /MDG* will not be generated again, but customer include structures are activated.</p>	

5	<p>In the following steps, you verify that your field extension was successfully added to the Customer Include of the structures.</p> <p>Start transaction SE11.</p> <p>Display structures /MDGBP/_S_BP_PDF_BP_CENTRL, /MDGBP/_S_BP_PP_BP_CENTRL, /MDGBP/_S_BP_ES_BP_CENTRL, /MDGBP/_S_BP_FP_BP_CENTRL by entering the details as shown.</p>																									
6	<p>The new field has been generated in the Customer Include of the mapping structure.</p>	 <table border="1" data-bbox="674 1089 1389 1302"> <thead> <tr> <th>Component</th> <th>Typing Method</th> <th>Component Type</th> <th>Data Type</th> </tr> </thead> <tbody> <tr> <td>TTLE_ROYL</td> <td>1 Types</td> <td>AD_TITLES</td> <td>CHAR</td> </tr> <tr> <td>XBLCK</td> <td>1 Types</td> <td>BU_XBLCK</td> <td>CHAR</td> </tr> <tr> <td>XDELE</td> <td>1 Types</td> <td>BU_XDELE</td> <td>CHAR</td> </tr> <tr> <td>.INCLUDE</td> <td>1 Types</td> <td>CI_MDG_S_PBP_BP_CE</td> <td>CHAR</td> </tr> <tr> <td>ZZLOB</td> <td>1 Types</td> <td>ZZ9_BUT000_LOB</td> <td>CHAR</td> </tr> </tbody> </table>	Component	Typing Method	Component Type	Data Type	TTLE_ROYL	1 Types	AD_TITLES	CHAR	XBLCK	1 Types	BU_XBLCK	CHAR	XDELE	1 Types	BU_XDELE	CHAR	.INCLUDE	1 Types	CI_MDG_S_PBP_BP_CE	CHAR	ZZLOB	1 Types	ZZ9_BUT000_LOB	CHAR
Component	Typing Method	Component Type	Data Type																							
TTLE_ROYL	1 Types	AD_TITLES	CHAR																							
XBLCK	1 Types	BU_XBLCK	CHAR																							
XDELE	1 Types	BU_XDELE	CHAR																							
.INCLUDE	1 Types	CI_MDG_S_PBP_BP_CE	CHAR																							
ZZLOB	1 Types	ZZ9_BUT000_LOB	CHAR																							

2.1.4. Define SMT Mapping

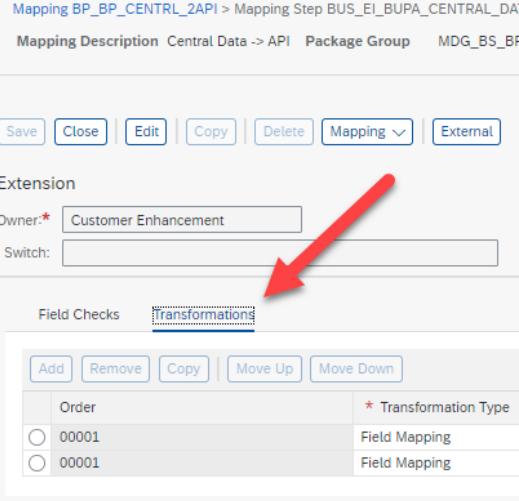
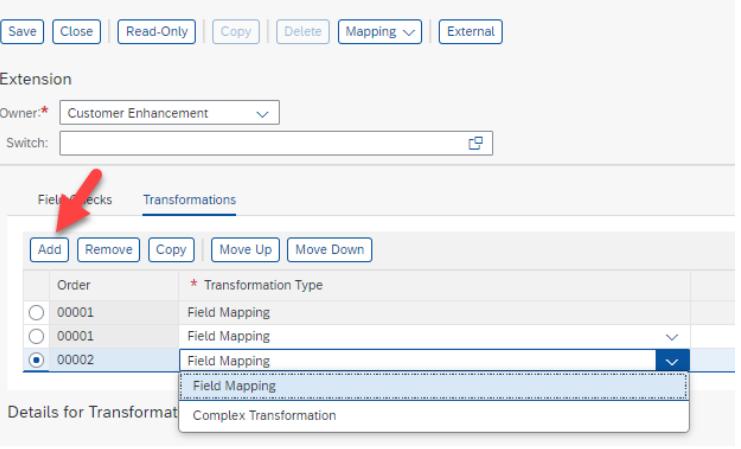
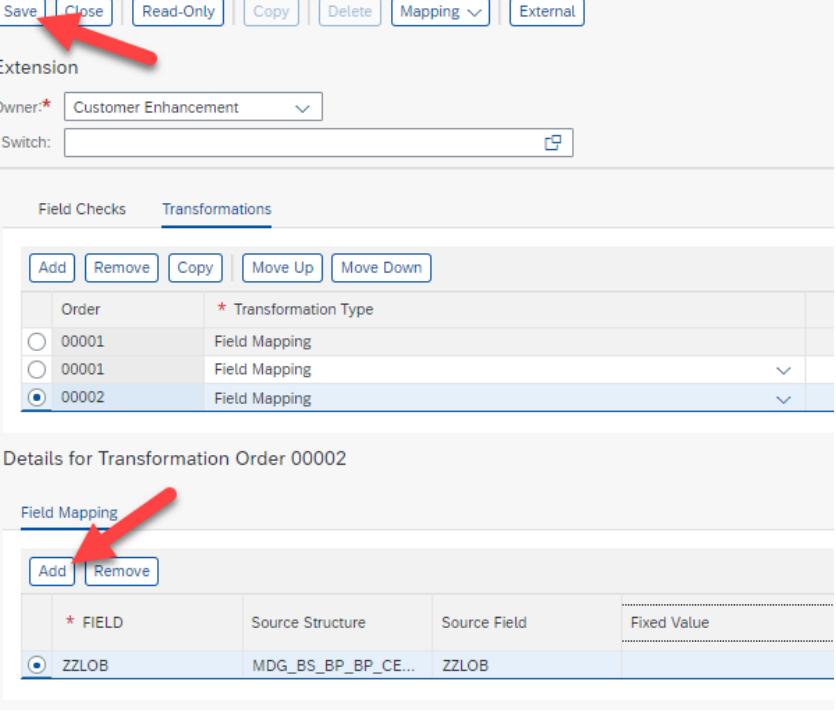
The Service Mapping Tool (SMT) is a program that enables you to fill target structures by using sets of source structures (in ABAP). It supports simple mappings, mappings with field transformations, complex transformations, and field checks.

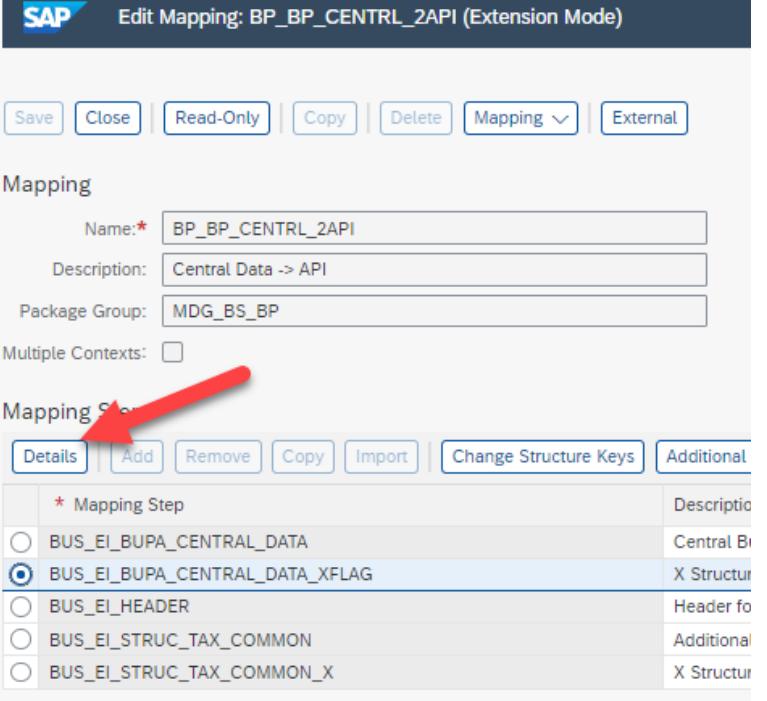
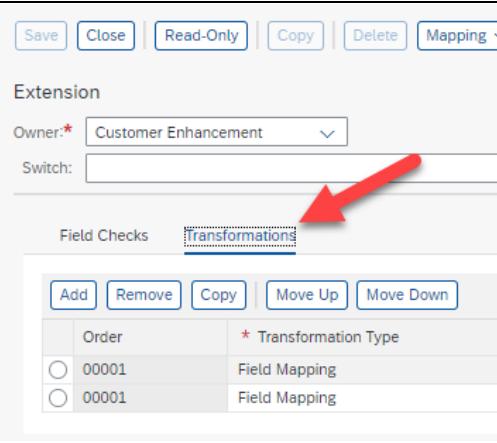
You need to perform the following mappings:

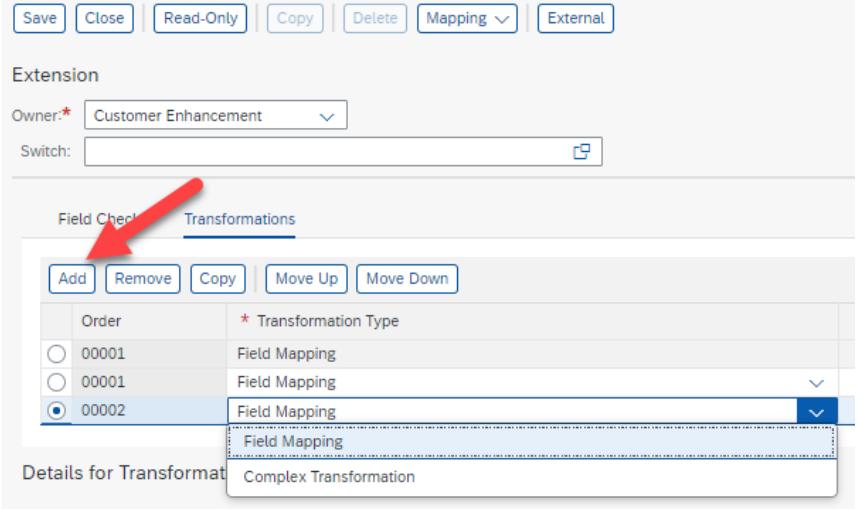
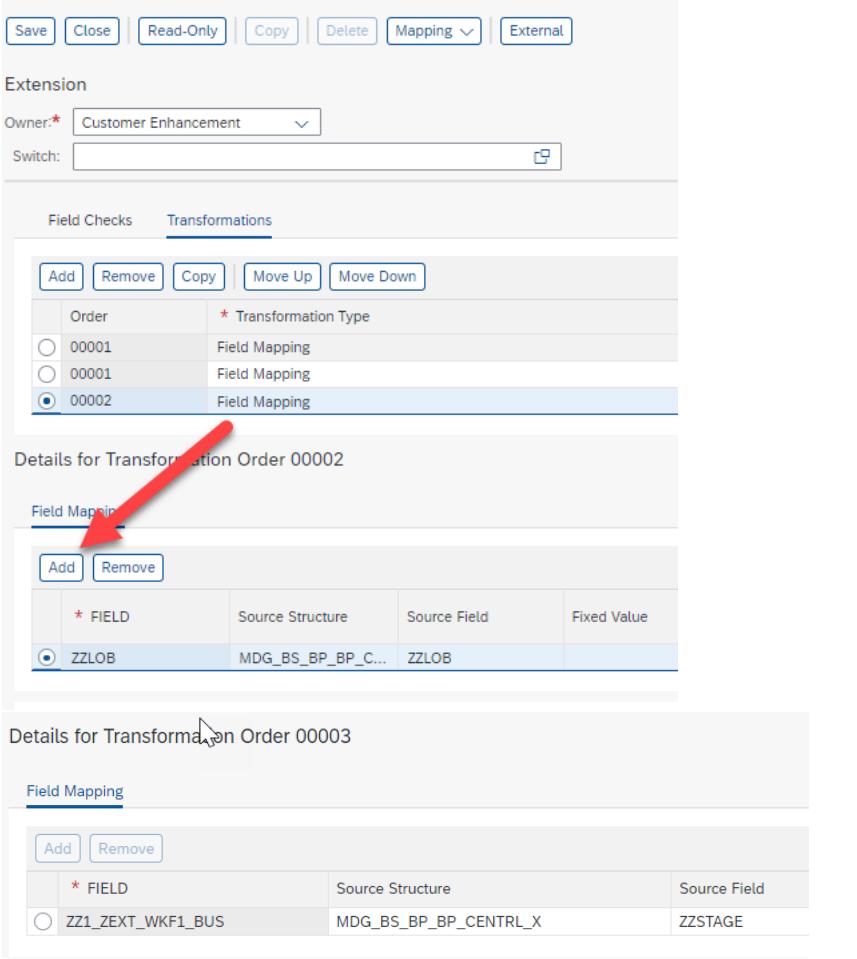
- Mapping Central Data to the API (BP_BP_CENTRL_2API)
 - Central Business Partner Data
 - X Structure for Central Business Partner Data
- Mapping Business Partner general data to MDG staging (MDG_BS_BP_MAP_2STA)
 - Central Business Partner Data

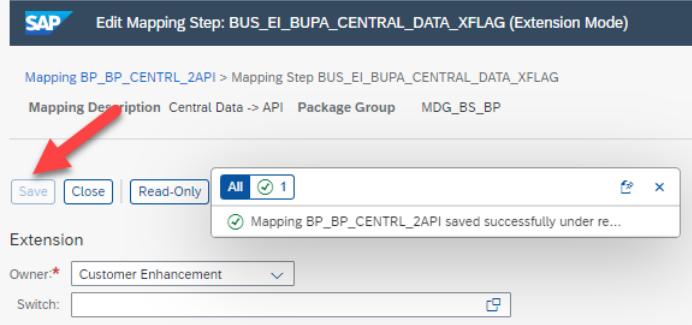
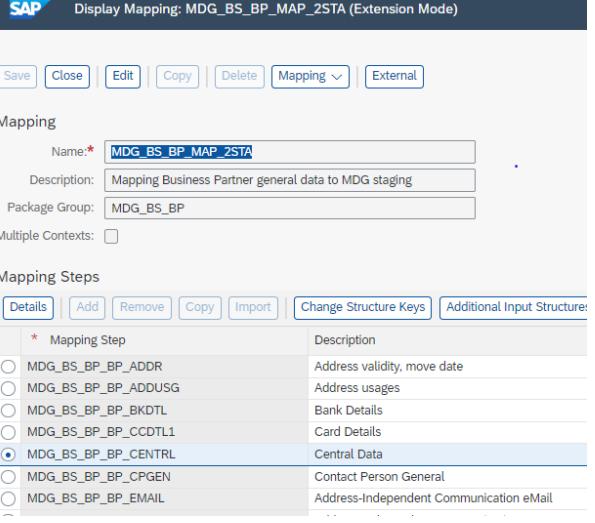
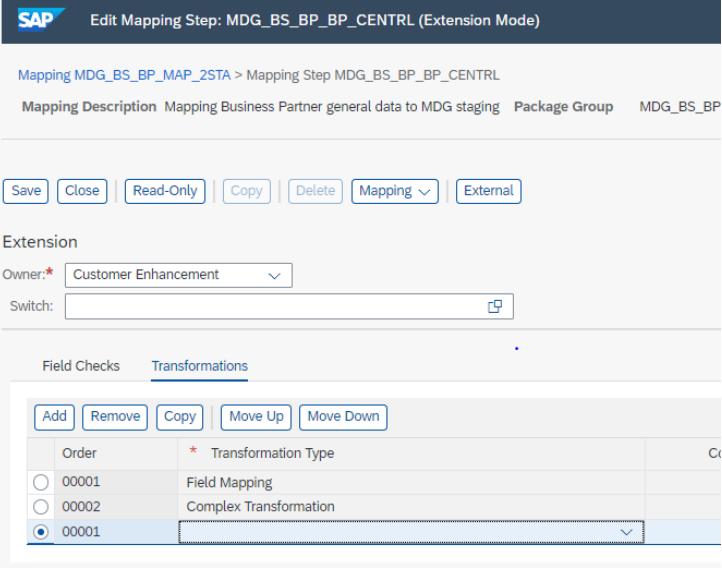
1	Log into system for cross-client maintenance.	
---	-----------------------------------------------	--

2	<p>Start Customizing for <i>Master Data Governance</i> (transaction MDGIMG).</p> <p>Choose <i>General Settings > Data Modeling > Extend Mappings > Extend Mappings</i>.</p> <p>Choose <i>Mapping > Open</i>.</p> <p>Enter BP_BP_CENTRL_2API.</p>																			
3	<p>You have opened the mapping of BP central data from MDG staging area to the service API.</p> <p>You see a table with mapping steps.</p> <p>You start by extending the mapping for the <i>Central Business Partner Data</i>.</p>	<table border="1"> <thead> <tr> <th>Mapping Step</th> <th>Description</th> <th>Source</th> </tr> </thead> <tbody> <tr> <td>BUS_EI_BUPA_CENTRAL_DATA</td> <td>Central Business Partner Data</td> <td>MDG_BS_E</td> </tr> <tr> <td>BUS_EI_BUPA_CENTRAL_DATA_XFLAG</td> <td>X Structure for Central Business Partner Data</td> <td>MDG_BS_E</td> </tr> <tr> <td>BUS_EI_HEADER</td> <td>Header for Business Partner Data</td> <td>MDG_BS_E</td> </tr> <tr> <td>BUS_EI_STRUC_TAX_COMMON</td> <td>Additional Data Tax Numbers</td> <td>MDG_BS_E</td> </tr> <tr> <td>BUS_EI_STRUC_TAX_COMMON_X</td> <td>X Structure for Additional Data Tax Numbers</td> <td>MDG_BS_E</td> </tr> </tbody> </table>	Mapping Step	Description	Source	BUS_EI_BUPA_CENTRAL_DATA	Central Business Partner Data	MDG_BS_E	BUS_EI_BUPA_CENTRAL_DATA_XFLAG	X Structure for Central Business Partner Data	MDG_BS_E	BUS_EI_HEADER	Header for Business Partner Data	MDG_BS_E	BUS_EI_STRUC_TAX_COMMON	Additional Data Tax Numbers	MDG_BS_E	BUS_EI_STRUC_TAX_COMMON_X	X Structure for Additional Data Tax Numbers	MDG_BS_E
Mapping Step	Description	Source																		
BUS_EI_BUPA_CENTRAL_DATA	Central Business Partner Data	MDG_BS_E																		
BUS_EI_BUPA_CENTRAL_DATA_XFLAG	X Structure for Central Business Partner Data	MDG_BS_E																		
BUS_EI_HEADER	Header for Business Partner Data	MDG_BS_E																		
BUS_EI_STRUC_TAX_COMMON	Additional Data Tax Numbers	MDG_BS_E																		
BUS_EI_STRUC_TAX_COMMON_X	X Structure for Additional Data Tax Numbers	MDG_BS_E																		
4	<p>Select the first mapping step as shown.</p> <p>Choose the <i>Details</i> pushbutton.</p>	<table border="1"> <thead> <tr> <th>Mapping Step</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>BUS_EI_BUPA_CENTRAL_DATA</td> <td>Central Business Partner Data</td> </tr> <tr> <td>BUS_EI_BUPA_CENTRAL_DATA_XFLAG</td> <td>X Structure for Central Business Partner Data</td> </tr> <tr> <td>BUS_EI_HEADER</td> <td>Header for Business Partner Data</td> </tr> </tbody> </table>	Mapping Step	Description	BUS_EI_BUPA_CENTRAL_DATA	Central Business Partner Data	BUS_EI_BUPA_CENTRAL_DATA_XFLAG	X Structure for Central Business Partner Data	BUS_EI_HEADER	Header for Business Partner Data										
Mapping Step	Description																			
BUS_EI_BUPA_CENTRAL_DATA	Central Business Partner Data																			
BUS_EI_BUPA_CENTRAL_DATA_XFLAG	X Structure for Central Business Partner Data																			
BUS_EI_HEADER	Header for Business Partner Data																			

5	<p>Select the <i>Transformations</i> tab.</p>	
6	<p>Change to edit mode by choosing the <i>Edit</i> pushbutton.</p> <p>Above the transformations table, choose the <i>Add</i> pushbutton to add a new transformation, assign your development package and choose <i>Field Mapping</i>.</p>	
7	<p>Choose the <i>Add</i> pushbutton above the <i>Field Mapping</i> table.</p> <p>Enter the mapping for ZZLOB/ZZ1_ZEXT_WKF1_BUS as shown.</p> <p>Save the changes.</p>	

		Details for Transformation Order 00002																																																	
		<p>Field Mapping</p> <table border="1"> <thead> <tr> <th>Add</th> <th>Remove</th> <th>* FIELD</th> <th>Source Structure</th> <th>Source Field</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="radio"/></td> <td></td> <td>ZZ1_ZEXT_WKF1_BUS</td> <td>MDG_BS_BP_BP_CENTRL</td> <td>ZZSTAGE</td> </tr> </tbody> </table>	Add	Remove	* FIELD	Source Structure	Source Field	<input checked="" type="radio"/>		ZZ1_ZEXT_WKF1_BUS	MDG_BS_BP_BP_CENTRL	ZZSTAGE																																							
Add	Remove	* FIELD	Source Structure	Source Field																																															
<input checked="" type="radio"/>		ZZ1_ZEXT_WKF1_BUS	MDG_BS_BP_BP_CENTRL	ZZSTAGE																																															
8	<p>Return to the <i>Mapping Step</i> table for the mapping <code>BP_BP_CENTRL_2API</code>.</p> <p>You have opened the mapping of BP central data from MDG staging area to the service API.</p> <p>You see a table with mapping steps.</p> <p>You will now extend the mapping for the Central Business Partner Data X structure.</p> <p>Select the <code>*_XFLAG</code> mapping step and choose the <i>Details</i> pushbutton.</p>	 <p>SAP Edit Mapping: BP_BP_CENTRL_2API (Extension Mode)</p> <p>Save Close Read-Only Copy Delete Mapping External</p> <p>Mapping</p> <p>Name: * BP_BP_CENTRL_2API</p> <p>Description: Central Data -> API</p> <p>Package Group: MDG_BS_BP</p> <p>Multiple Contexts: <input type="checkbox"/></p> <p>Mapping Steps</p> <table border="1"> <thead> <tr> <th>Details</th> <th>Add</th> <th>Remove</th> <th>Copy</th> <th>Import</th> <th>Change Structure Keys</th> <th>Additional</th> </tr> </thead> <tbody> <tr> <td colspan="7">* Mapping Step</td> </tr> <tr> <td><input type="radio"/></td> <td>BUS_EI_BUPA_CENTRAL_DATA</td> <td></td> <td></td> <td></td> <td></td> <td>Central Bi</td> </tr> <tr> <td><input checked="" type="radio"/></td> <td>BUS_EI_BUPA_CENTRAL_DATA_XFLAG</td> <td></td> <td></td> <td></td> <td></td> <td>X Structur</td> </tr> <tr> <td><input type="radio"/></td> <td>BUS_EI_HEADER</td> <td></td> <td></td> <td></td> <td></td> <td>Header fo</td> </tr> <tr> <td><input type="radio"/></td> <td>BUS_EI_STRUC_TAX_COMMON</td> <td></td> <td></td> <td></td> <td></td> <td>Additional</td> </tr> <tr> <td><input type="radio"/></td> <td>BUS_EI_STRUC_TAX_COMMON_X</td> <td></td> <td></td> <td></td> <td></td> <td>X Structur</td> </tr> </tbody> </table>	Details	Add	Remove	Copy	Import	Change Structure Keys	Additional	* Mapping Step							<input type="radio"/>	BUS_EI_BUPA_CENTRAL_DATA					Central Bi	<input checked="" type="radio"/>	BUS_EI_BUPA_CENTRAL_DATA_XFLAG					X Structur	<input type="radio"/>	BUS_EI_HEADER					Header fo	<input type="radio"/>	BUS_EI_STRUC_TAX_COMMON					Additional	<input type="radio"/>	BUS_EI_STRUC_TAX_COMMON_X					X Structur
Details	Add	Remove	Copy	Import	Change Structure Keys	Additional																																													
* Mapping Step																																																			
<input type="radio"/>	BUS_EI_BUPA_CENTRAL_DATA					Central Bi																																													
<input checked="" type="radio"/>	BUS_EI_BUPA_CENTRAL_DATA_XFLAG					X Structur																																													
<input type="radio"/>	BUS_EI_HEADER					Header fo																																													
<input type="radio"/>	BUS_EI_STRUC_TAX_COMMON					Additional																																													
<input type="radio"/>	BUS_EI_STRUC_TAX_COMMON_X					X Structur																																													
9	Select the <i>Transformations</i> tab.	 <p>Save Close Read-Only Copy Delete Mapping</p> <p>Extension</p> <p>Owner: * Customer Enhancement</p> <p>Switch:</p> <p>Field Checks Transformations</p> <table border="1"> <thead> <tr> <th>Add</th> <th>Remove</th> <th>Copy</th> <th>Move Up</th> <th>Move Down</th> </tr> </thead> <tbody> <tr> <td colspan="2">Order</td> <td>* Transformation Type</td> <td colspan="2"></td> </tr> <tr> <td><input type="radio"/></td> <td>00001</td> <td>Field Mapping</td> <td colspan="2"></td> </tr> <tr> <td><input type="radio"/></td> <td>00001</td> <td>Field Mapping</td> <td colspan="2"></td> </tr> </tbody> </table>	Add	Remove	Copy	Move Up	Move Down	Order		* Transformation Type			<input type="radio"/>	00001	Field Mapping			<input type="radio"/>	00001	Field Mapping																															
Add	Remove	Copy	Move Up	Move Down																																															
Order		* Transformation Type																																																	
<input type="radio"/>	00001	Field Mapping																																																	
<input type="radio"/>	00001	Field Mapping																																																	

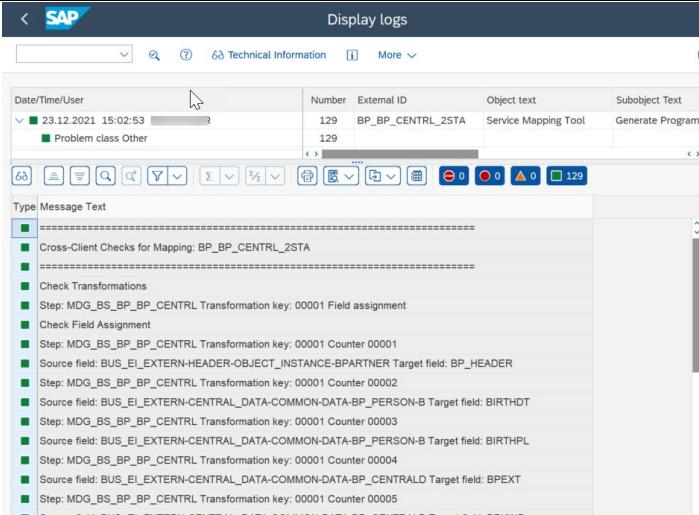
	<p>10 Change to edit mode by choosing the <i>Edit</i> pushbutton.</p> <p>Above the <i>Transformations</i> table choose the <i>Add</i> pushbutton to add a new transformation and select <i>Field Mapping</i>.</p> 
11	<p>Choose the <i>Add</i> pushbutton above the <i>Field Mapping</i> table and enter the mapping for ZZLOB / ZZ1_ZEXT_WKF1_BUS as shown.</p> 

12	<p>Save the changes.</p>	
13	<p>Select <i>Mapping</i> -> <i>Open</i>.</p> <p>Enter MDG_BS_BP_MAP_2STA. You have opened the mapping of BP general data from the service API to the MDG staging area.</p> <p>You see a table with mapping steps.</p> <p>You need to extend the mapping for the <i>Central Data</i>. Choose the <i>Details</i> pushbutton.</p>	
14	<p>Select the <i>Transformations</i> tab.</p> <p>Choose the <i>Edit</i> pushbutton.</p> <p>Select the package.</p> <p>Choose the <i>Add</i> pushbutton to add a new transformation and select <i>Field Mapping</i>.</p>	

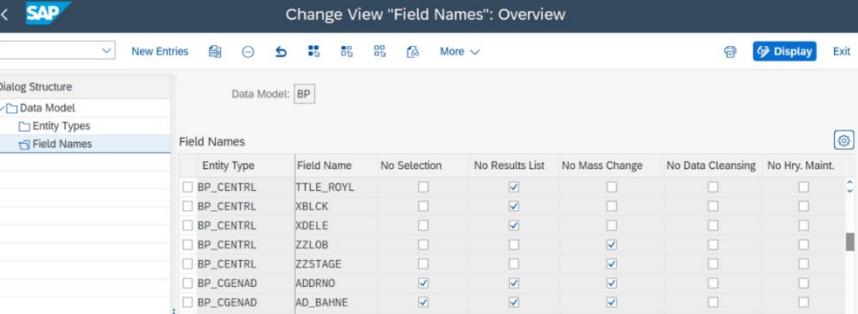
	<p>15 Choose the <i>Add</i> pushbutton above the <i>Field Mapping</i> table.</p> <p>Enter the mapping for ZZLOB/ZZ1_ZEXT_WKF1_BUS as shown.</p> <p>Save your changes.</p>	
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

2.1.5. Check Customizing

1	<p>Start transaction MDGIMG. Navigate to <i>General Settings > Data Modeling > Create and Edit Mappings > Check Customizing</i>.</p>
2	<p>Enter mapping BP_BP_CENTRL_2API (you might use the value help). Choose <i>Execute</i>. Check that no error message is issued.</p>

3	<p>Repeat this for mapping BP_BP_CENTRL_2STA.</p> <p>Check that no error message is issued.</p>  <p>The screenshot shows the SAP Display logs interface. At the top, it says "Display logs". Below that is a table with columns: Date/Time/User, Number, External ID, Object text, and Subobject Text. A log entry is shown: "23.12.2021 15:02:53 129 BP_BP_CENTRL_2STA Service Mapping Tool Generate Program". The "Object text" column contains the log message. Below the table is a list of steps: "Cross-Client Checks for Mapping: BP_BP_CENTRL_2STA", "Check Transformations", "Step: MDG_BS_BP_BP_CENTRL Transformation key: 00001 Field assignment", "Step: MDG_BS_BP_BP_CENTRL Transformation key: 00001 Counter 00001", "Source field: BUS_EI_EXTERN-HEADER-OBJECT_INSTANCE-BPARTNER Target field: BP_HEADER", "Step: MDG_BS_BP_BP_CENTRL Transformation key: 00001 Counter 00002", "Source field: BUS_EI_EXTERN-CENTRAL_DATA-COMMON-DATA-BP_PERSON-B Target field: BIRTHDT", "Step: MDG_BS_BP_BP_CENTRL Transformation key: 00001 Counter 00003", "Source field: BUS_EI_EXTERN-CENTRAL_DATA-COMMON-DATA-BP_PERSON-B Target field: BIRTHPL", "Step: MDG_BS_BP_BP_CENTRL Transformation key: 00001 Counter 00004", "Source field: BUS_EI_EXTERN-CENTRAL_DATA-COMMON-DATA-BP_CENTRALD Target field: BPEXT", and "Step: MDG_BS_BP_BP_CENTRL Transformation key: 00001 Counter 00005".</p>
---	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.1.6. Define Field Properties for UI

1	Log into system for cross-client maintenances.	
2	<p>Start transaction MDGIMG.</p> <p>Navigate to General Settings > UI Modeling > Define Field Properties for UI.</p> <p>Select data model <i>BP</i> and choose <i>Field Names</i>.</p> <p>Add the new fields for entity <i>BP_CENTRL</i>.</p> <p>Save your changes.</p>	 <p>The screenshot shows the SAP Change View "Field Names": Overview interface. On the left, there's a tree view with "Dialog Structure", "Data Model" (selected), "Entity Types", and "Field Names". The main area is titled "Field Names" with "Data Model: BP". It shows a table with columns: Entity Type, Field Name, No Selection, No Results List, No Mass Change, No Data Cleansing, and No Hr. Maint. The table lists fields for BP_CENTRL: TTLE_ROYL, XBLCK, XDELE, ZZLOB, ZZSTAGE, ADDRNO, and AD_BAHNE. Most fields have "No Selection" checked, while ADDRNO has it unchecked. Other checkboxes like "No Results List" and "No Mass Change" are also checked for some fields.</p>

2.1.7. Custom Handler Class

A custom handler class is necessary if you have custom fields and you want to use the file download application or the query method.

Ensure these SAP Notes already implemented:

- 3223641 MDG-BP/C/S: Invalid execution of the Query Methods of SAP owned Handler Classes
- 3223718 MDG-BP/C/S: Type-casted Mapping for SAP owned Handler Classes (1)
- 3223776 MDG-BP/C/S: Type-casted Mapping for SAP owned Handler Classes (2)

To create a custom handler class follow the guide here: [Create and Register a Custom Handler Class](#) on the MDG Community [How-to Information for SAP Master Data Governance](#).

1 Log into system for cross-client maintenances and follow the instructions in the above mentioned guide to create a custom handler class for example ZCL_CUSTOM_BP_HANDLER (inheritance from CL_MDG_BS_FND_HANDLER)					
2 Redefine method IF_MDG_BS_BP_ACCESS_HANDLER~EXECUTE_QUERY. See Appendix 564.1 Example coding for class ZCL_CUSTOM_BP_HANDLER.					
3 Specify handler classes for your own data to be processed by Master Data Governance in MDGIMG > Central Governance for Business Partner > Specify Handler Processed by Access Class.	<p>Maintenance of handler processed by access class</p> <table border="1"> <thead> <tr> <th>Data Model</th> <th>Class/Interface</th> </tr> </thead> <tbody> <tr> <td>BP</td> <td>ZCL_CUSTOM_BP_HANDLER</td> </tr> </tbody> </table>	Data Model	Class/Interface	BP	ZCL_CUSTOM_BP_HANDLER
Data Model	Class/Interface				
BP	ZCL_CUSTOM_BP_HANDLER				

2.1.8. Define Field Mapping for SAP HANA-Based Search

To use SAP HANA-based search as a search provider, search views are created using the Create Search View function. For data model BP you need to define the mapping between the MDG model entities and attributes, and the reuse attributes.

More information for Configuring SAP HANA-Based Search for MDG:

<https://help.sap.com/viewer/6d52de87aa0d4fb6a90924720a5b0549/latest/en-US/7293f8516599a060e10000000a44176d.html>

1 Log into system for cross-client maintenances.	
2 Start transaction MDGIMG. Navigate to General Settings > Data Quality and Search > Search and Duplicate Check > Define Joins, Field Mapping, and Authorizations for Reuse Tables. Choose Field Mapping and choose New	

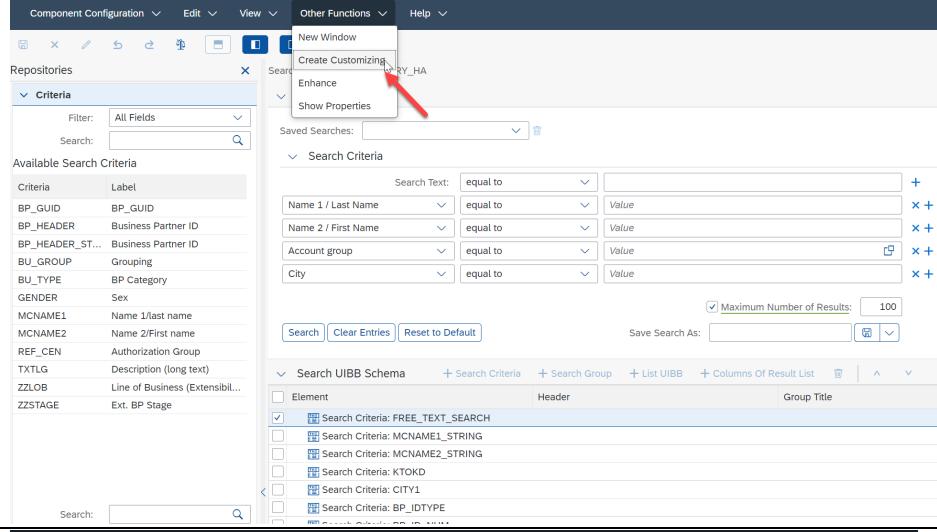
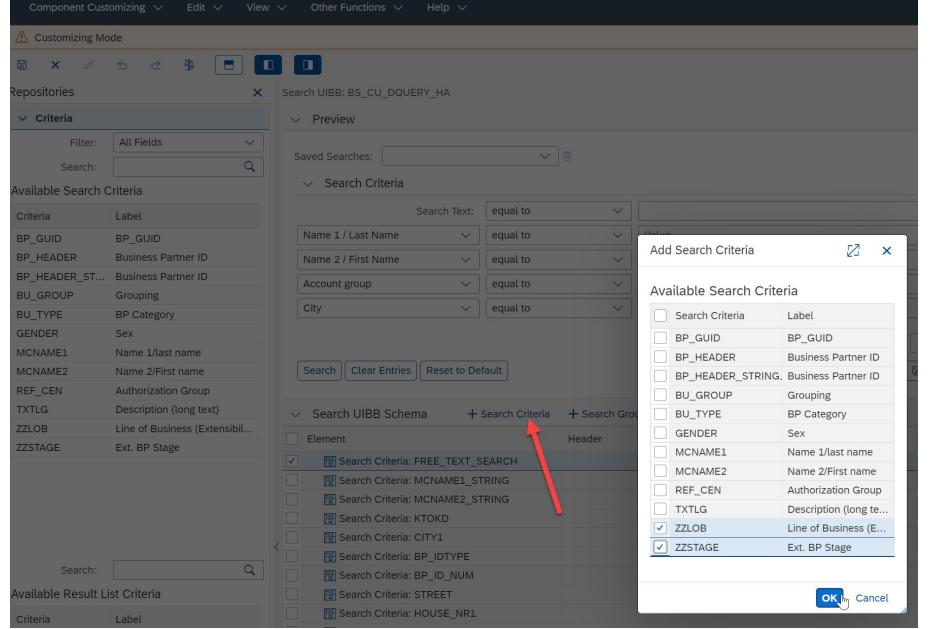
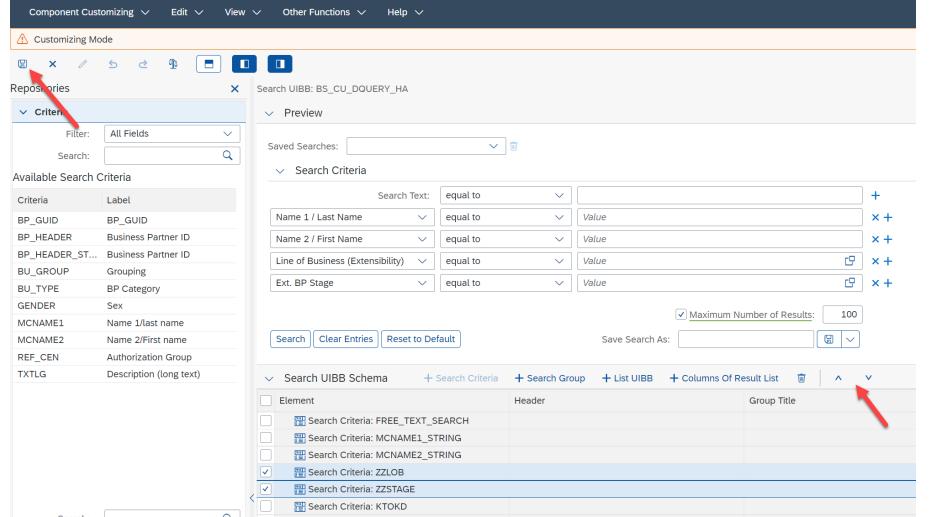
	<p>Entries. Add the new fields for entity BP_CENTRL. Save your changes.</p>																						
3	<p>Start transaction MDGIMG. Navigate to General Settings > Data Quality and Search > Search and Duplicate Check > Create Search View.</p> <p>Add the new fields to your search view in step 2 Select Entities and Attributes. Review and Generate.</p>	<p>Search View :: MDG_BS_CUS_CUSTOMER Description :: Customer Search (HANA) Business Object Type</p> <p>Review and Generate</p> <p>Entities and Attributes</p> <ul style="list-style-type: none"> BP - Business Partner <ul style="list-style-type: none"> BP_HEADER - Business Partner ID <ul style="list-style-type: none"> Attributes ADDRESS - Address BP_CENTRL - Central Data <ul style="list-style-type: none"> Attributes <ul style="list-style-type: none"> BU_SORT1 - Search term 1 GENDER - Sex MCNAME1 - Name 1/last name MCNAME2 - Name 2/First name ZZLOB - Line of Business (Extensibility) ZZSTAGE - Ext. BP Stage REF_CEN - Authorization Group BP_IDNUM - Identification Numbers BP_MLT_AS - Multiple Assignment of Customer/Supplier 																					
4	<p>Optional: Start transaction MDGIMG. Navigate to General Settings > Data Quality and Search > Search and Duplicate Check > Define Search Applications. Choose Search Mode HA, choose search help MDG_BS_CUS_CUSTOMER and Allocation of Fuzziness of Attribute in SearchHelp.</p>	<table border="1"> <thead> <tr> <th>Data Model</th> <th>Included search help</th> <th>Description (medium text)</th> </tr> </thead> <tbody> <tr> <td>0G</td> <td>MDGF_0G_SUBMPACK</td> <td>HANA Search</td> </tr> <tr> <td>0G</td> <td>MDGF_0G_TRANSTYPE</td> <td>HANA Search</td> </tr> <tr> <td>BP</td> <td>MDG_BS_BP_ADDRESS</td> <td>Address Search (HANA)</td> </tr> <tr> <td>BP</td> <td>MDG_BS_BP_PERSON</td> <td>Person Search (HANA)</td> </tr> <tr> <td><input checked="" type="checkbox"/> BP</td> <td>MDG_BS_CUS_CUSTOMER</td> <td>Customer Search (HANA)</td> </tr> <tr> <td>BP</td> <td>MDG_BS_SUP_SUPPLIER</td> <td>Supplier Search (HANA)</td> </tr> </tbody> </table>	Data Model	Included search help	Description (medium text)	0G	MDGF_0G_SUBMPACK	HANA Search	0G	MDGF_0G_TRANSTYPE	HANA Search	BP	MDG_BS_BP_ADDRESS	Address Search (HANA)	BP	MDG_BS_BP_PERSON	Person Search (HANA)	<input checked="" type="checkbox"/> BP	MDG_BS_CUS_CUSTOMER	Customer Search (HANA)	BP	MDG_BS_SUP_SUPPLIER	Supplier Search (HANA)
Data Model	Included search help	Description (medium text)																					
0G	MDGF_0G_SUBMPACK	HANA Search																					
0G	MDGF_0G_TRANSTYPE	HANA Search																					
BP	MDG_BS_BP_ADDRESS	Address Search (HANA)																					
BP	MDG_BS_BP_PERSON	Person Search (HANA)																					
<input checked="" type="checkbox"/> BP	MDG_BS_CUS_CUSTOMER	Customer Search (HANA)																					
BP	MDG_BS_SUP_SUPPLIER	Supplier Search (HANA)																					

5 You can add an <i>Entity Type</i> and <i>Attribute</i> and then specify the <i>Fuzziness</i> you would like for non-key fields. The fuzziness threshold ranges from greater than 0.7 to less than or equal to 1.0. The default value for fuzziness is 0.7 for non-key fields when you have not maintained an entry in the customizing table, excluding special cases. Defining a fuzziness threshold of 1.0 means that only exact matches will be returned in the search results when using the <i>EQUAL TO</i> operator.	<table border="1"> <thead> <tr> <th>Entity Type</th> <th>Attribute</th> <th>Fuzziness</th> </tr> </thead> <tbody> <tr> <td>BP_CENTRL</td> <td>ZZLOB</td> <td>1,00</td> </tr> <tr> <td>BP_CENTRL</td> <td>ZZSTAGE</td> <td>1,00</td> </tr> </tbody> </table>	Entity Type	Attribute	Fuzziness	BP_CENTRL	ZZLOB	1,00	BP_CENTRL	ZZSTAGE	1,00
Entity Type	Attribute	Fuzziness								
BP_CENTRL	ZZLOB	1,00								
BP_CENTRL	ZZSTAGE	1,00								

2.1.9. Enhance Search UIBB

In this Customizing activity, you add the custom fields as selection criteria to the search UI. The configuration ID for search method Customer Search (HANA) is BS_CU_DQUERY_HA, the configuration ID for search method Supplier Search (HANA) is BS_SP_DQUERY_HA.

1 Log into system for cross-client maintenances.	
2 Start SE80 for Web Dynpro component: FPM_SEARCH_UIBB and open Component Configurations.	
3 Select config ID BS_CU_DQUERY_HA and choose Start Configurator and choose Display.	

4 Choose <i>Create Customizing</i> , enter a description and choose a transport.	
5 Choose <i>+Search Criteria</i> and select your custom fields. Move the fields to the desired place. Save your customizing.	
6 Move the fields to the desired place. Save your customizing.	

7 Check the Search Customers UI for Search Method: Customer Search (HANA).	<p>The screenshot shows the SAP Search Customers interface. At the top, it says "Search Method: Customer Search (HANA)". Below that is a "Search Criteria" section with four search fields: "Name 1 / Last Name", "Name 2 / First Name", "Line of Business (Extensibility)", and "Ext. BP Stage", each with dropdown menus and operators like "equal to" and "Value". There is also a "Maximum Number of Results" field set to 100. Below the search criteria is a "Result List" section with various buttons like "New", "Block", "Change Documents", "Replication Status", "Replicate", "Create Cleansing Case", "Copy", and "Multi-Processing". A note below the list says "The table does not contain any data".</p>
----------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.1.10. Adjust Staging Area of Linked Change Requests

1	Log into system for which change request adjustment is required.	
2	Start Customizing for Master Data Governance (transaction MDGIMG). Select General Settings> Data Modeling> Edit Data Model. Select data model <i>BP</i> and choose the button <i>Adjust Staging Area of Linked Change Requests</i> as shown in the screenshot.	<p>The screenshot shows the SAP Inactive Data Models overview screen. It has a navigation bar with "Table View", "Process", "Goto", "Selection", "Utilities", "System", and "Hilfe". Below the navigation is a title "Display View 'Inactive Data Models': Overview". On the left is a tree view with "Inactive Data Models", "Entity Types", "Relationships", and "Fields of Foreign Key Relation". The main area shows a table of data models with columns "Data Model", "Description (medium text)", "ActiveArea", and "Prefix/". The data models listed are OF, OG, BP (selected), MM, Chart of Accounts, Financials, Business Partner, and Material. A red arrow points to the "More" button in the top right corner of the screen.</p>
3	Close the pop-up.	<p>The screenshot shows a pop-up window titled "Inactive Data Models" with a table. The table has columns "Data Model" and "Description". The row for "BP" is selected, and its description "Business Partner" is visible. A message at the bottom right says "Conversion program USMD_ADJUST_STAGING scheduled in client 9".</p>

2.1.11. Test Your Configuration

To test your configuration you need to add your new fields to the MDG single maintenance UI. Section 2.4 *User Interface Configuration* explains how to add the new fields to the UI.

Create a new change request in MDG and enter values for the new fields. Then process the change request until it is activated. Write down the ID of the new customer after activation.

SAP Organization: 9118, Org GmbH / Freiburg

Refresh Check Duplicate Check Change Documents Workflow Log

Customer Details Edit

General Data		Address	
Business Partner ID / Grouping:	9118	Internal Number Assignment:	<input type="button" value="▼"/>
Title:	<input type="button" value="▼"/>		
Name 1:*	Org GmbH		
Name 2:	<input type="button" value="▼"/>		
Salutation:	<input type="button" value="▼"/>		
Search Terms:	SALES	<input type="button" value="▼"/>	<input type="button" value="▼"/>
Organizational Data		Standard Communication	
Legal Entity:	<input type="button" value="▼"/>		
Legal Form:	<input type="button" value="▼"/>		
Date Founded:	<input type="button" value="▼"/>		
Liquidation Date:	<input type="button" value="▼"/>		
Authorization Group:	Stakeholder: Visibility 0 (Unrestricted)		
International Location Number:	<input type="button" value="▼"/>	<input type="button" value="▼"/>	<input type="button" value="▼"/>
Custom Additional Data		Telephone (Country/Region, Number, Extension):	
Line of Business:	Logistic EMEA		
External BP Stage:	101		
PO Box Postal Code / PO Box:	<input type="button" value="▼"/>		
PO Box Without Number:	<input type="checkbox"/>		
Company Postal Code:	<input type="button" value="▼"/>		
Mobile (Country/Region, Number):	<input type="button" value="▼"/>	<input type="button" value="▼"/>	<input type="button" value="▼"/>
Fax (Country/Region, Number, Extension):	<input type="button" value="▼"/>	<input type="button" value="▼"/>	<input type="button" value="▼"/>
Email:	<input type="button" value="▼"/>		
Website:	<input type="button" value="▼"/>		

Run transaction SE16 and find your newly created Business Partner in table BUT000. Make sure the fields ZZLOB and ZZ1_ZEXT_WKF1_BUS is there and contain the right value.

2.2. Extended Field Exists in Vendor / Customer Only

In this section you will learn how to add a new field **License** to the MDG BP data model. In the figure below you can see that it is necessary to extend the MDG BP data model. You also must have extended the Vendor / Customer table on the MDG hub and client systems. In order to facilitate data transfer between the MDG staging area and the LFA1 / KNA1 table on the MDG hub you have to extend the SMT mapping.

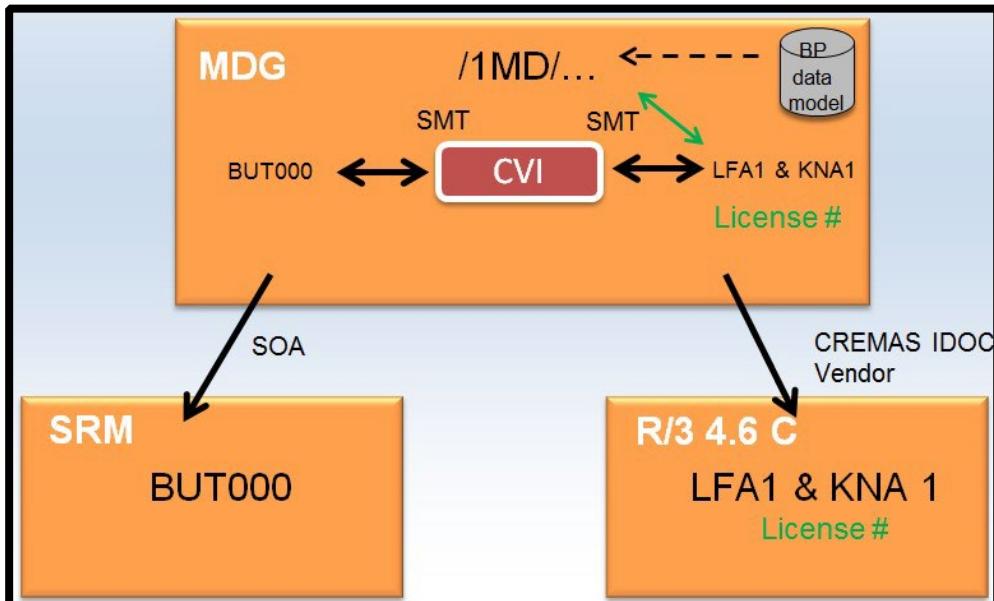


Figure: Field is added only to Vendor / Customer

This section is subdivided in to one sub-section for each implementation step. Take a look at the table below to get an overview of the steps involved.

Implementation Steps	Field only in Business Partner	Field only in Vendor/Customer	Field in both BP and Vendor/ Customer
Extend ERP data model	✓	✓	✓
Extend MDG data model	✓	✓	✓
Generate MDG data model	✓	✓	✓
Define SMT mapping	✓ ¹	✓ ²	✓ ¹
Implement CVI mapping			✓

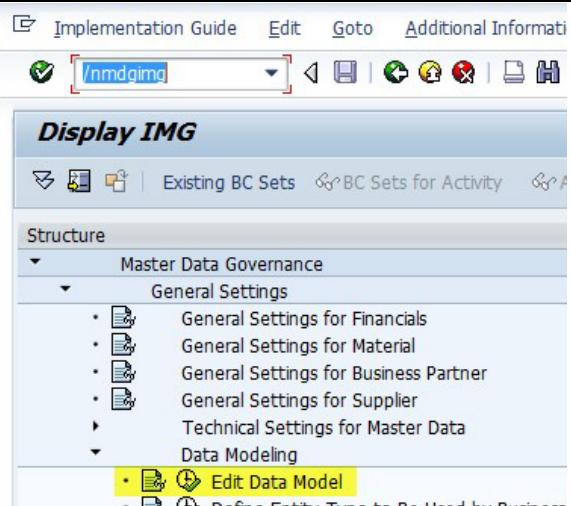
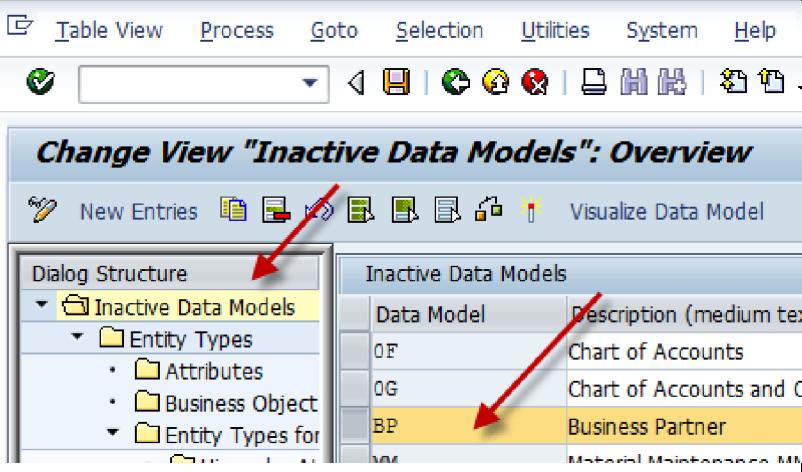
SMT Mappings
¹ MDG staging ↔ Business Partner
² MDG staging ↔ Vendor / Customer

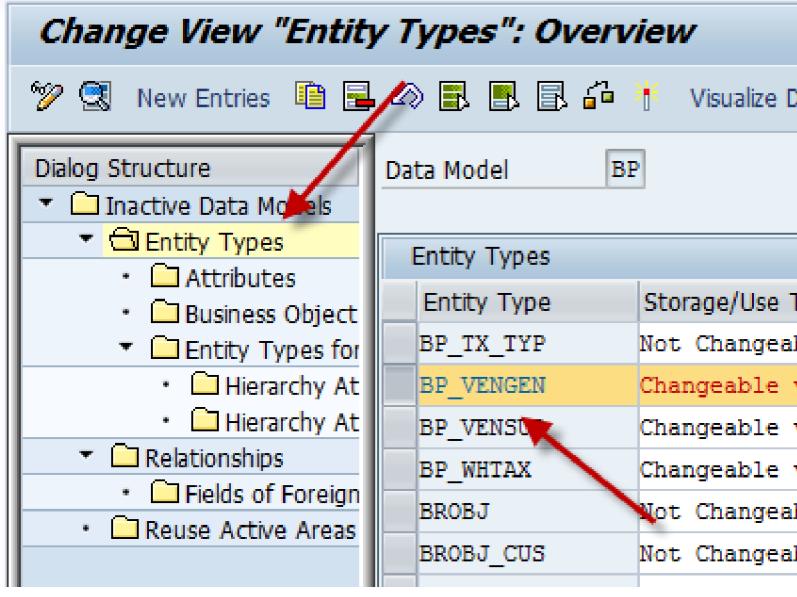
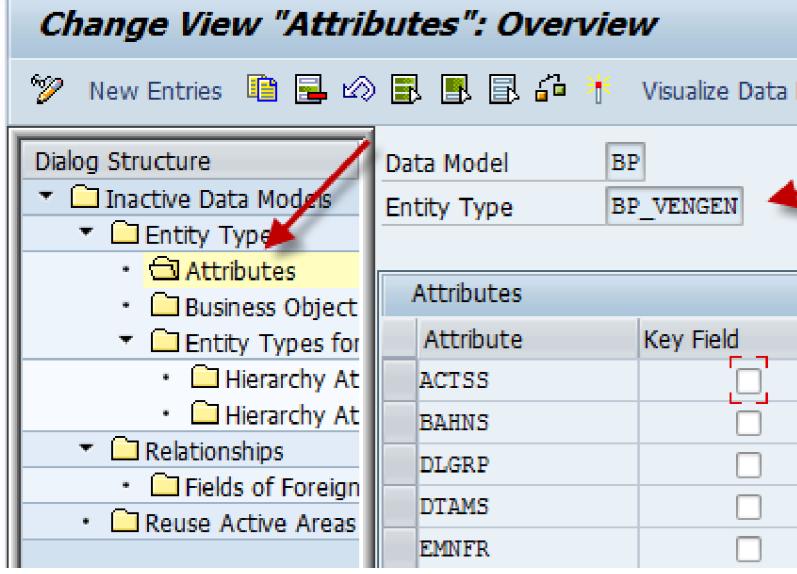
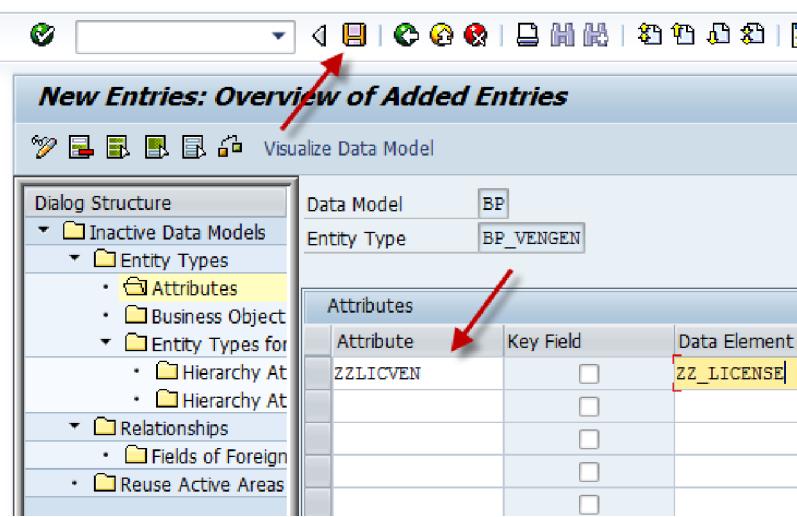
Since this scenario contains several similarities to the one explained in section Extended Field Exists in Business Partner and the one explained in section Extended Field Exists in Vendor / Customer on the detailed instructions below focus only on the additional steps.

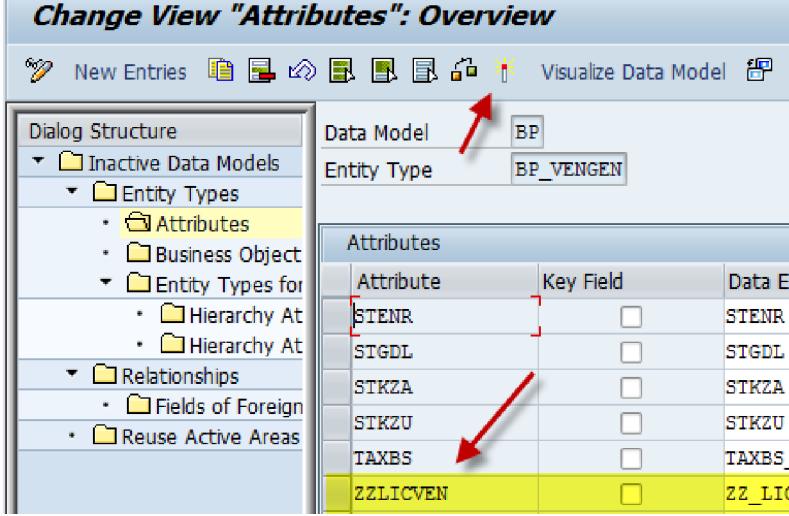
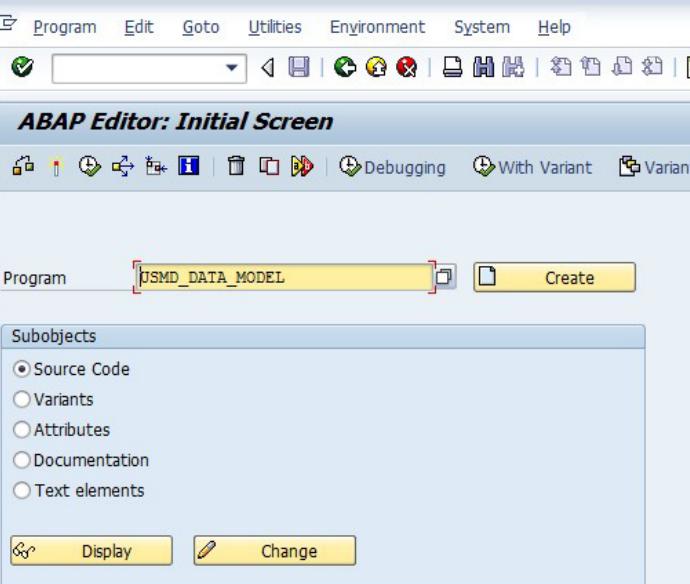
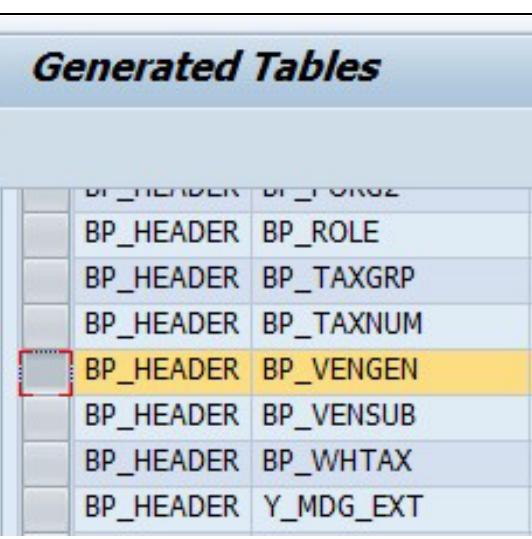
Extend ERP data model	For the extension of LFA1 follow the steps in section Extend the BUT000 Data Model above.
Extend MDG data model	To extend the MDG data model, follow the steps below.

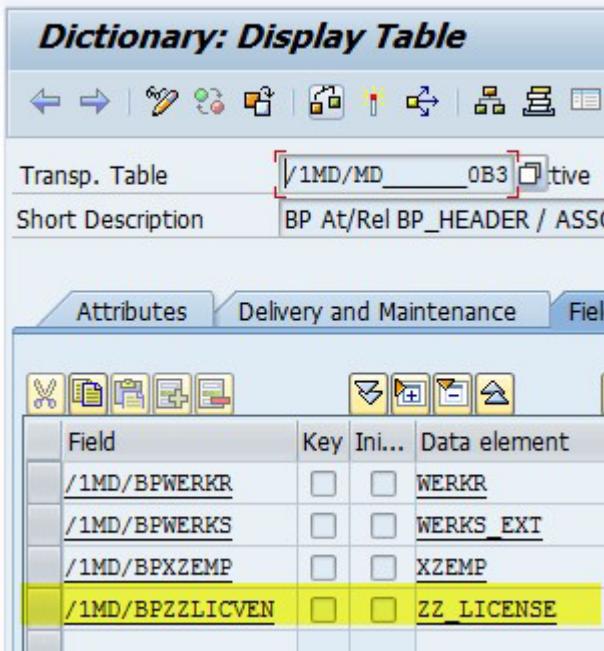
Generate MDG data model specific structures	To generate the MDG data model, follow the steps below.
Adjust Staging Area of Linked Change Requests	Follow steps in the section <i>Adjust Staging Area of Linked Change Requests</i> above.
Define SMT mapping	To define SMT mapping between MDG staging and vendor, follow the steps below.

2.2.1. Extend the MDG Data Model

1	Log into system for cross-client maintenances.	
2	Start Customizing for Master Data Governance (transaction MDGIMG). Select General Settings -> Data Modeling -> Edit Data Model .	
3	Select data model BP .	

4	<p>Double-click entity types. Select entity BP_VENGEN.</p> 
5	<p>Double-click Attributes. Choose New Entries to add a new attribute to the entity BP_VENGEN.</p> 
6	<p>Enter the field name ZZLICVEN and an appropriate data element (for example, ZZ_LICENSE).</p> 

7. Activate the changes.	
8 Start transaction SE38. Run program USMD_DATA_MODEL.	
9 Locate entity type BP_VENGEN and double-click it to see the generated table.	

<p>10. Verify that the generated table contains the added field /1MD/BPZZLICVEN.</p>	 <p>The screenshot shows the SAP ABAP Dictionary: Display Table interface. At the top, there are various toolbar icons. Below the toolbar, the table name is displayed as 'Transp. Table' followed by the table name itself, which is highlighted with a red box. The 'Short Description' is also visible. The table has three tabs at the bottom: 'Attributes', 'Delivery and Maintenance' (which is selected), and 'Field'. The 'Delivery and Maintenance' tab shows several icons for managing the table. The main area displays a grid of fields. One row is highlighted with yellow background and green border, corresponding to the newly added field. The columns in the grid are labeled 'Field', 'Key', 'Ini...', and 'Data element'. The data for the highlighted row is as follows:</p> <table border="1"> <thead> <tr> <th>Field</th><th>Key</th><th>Ini...</th><th>Data element</th></tr> </thead> <tbody> <tr> <td>/1MD/BPWERKR</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td>WERKR</td></tr> <tr> <td>/1MD/BPWERKS</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td>WERKS_EXT</td></tr> <tr> <td>/1MD/BPXZEMP</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td>XZEMP</td></tr> <tr style="background-color: #ffffcc; border: 2px solid #ffcc00;"> <td>/1MD/BPZZLICVEN</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td>ZZ_LICENSE</td></tr> </tbody> </table>	Field	Key	Ini...	Data element	/1MD/BPWERKR	<input type="checkbox"/>	<input type="checkbox"/>	WERKR	/1MD/BPWERKS	<input type="checkbox"/>	<input type="checkbox"/>	WERKS_EXT	/1MD/BPXZEMP	<input type="checkbox"/>	<input type="checkbox"/>	XZEMP	/1MD/BPZZLICVEN	<input type="checkbox"/>	<input type="checkbox"/>	ZZ_LICENSE
Field	Key	Ini...	Data element																		
/1MD/BPWERKR	<input type="checkbox"/>	<input type="checkbox"/>	WERKR																		
/1MD/BPWERKS	<input type="checkbox"/>	<input type="checkbox"/>	WERKS_EXT																		
/1MD/BPXZEMP	<input type="checkbox"/>	<input type="checkbox"/>	XZEMP																		
/1MD/BPZZLICVEN	<input type="checkbox"/>	<input type="checkbox"/>	ZZ_LICENSE																		

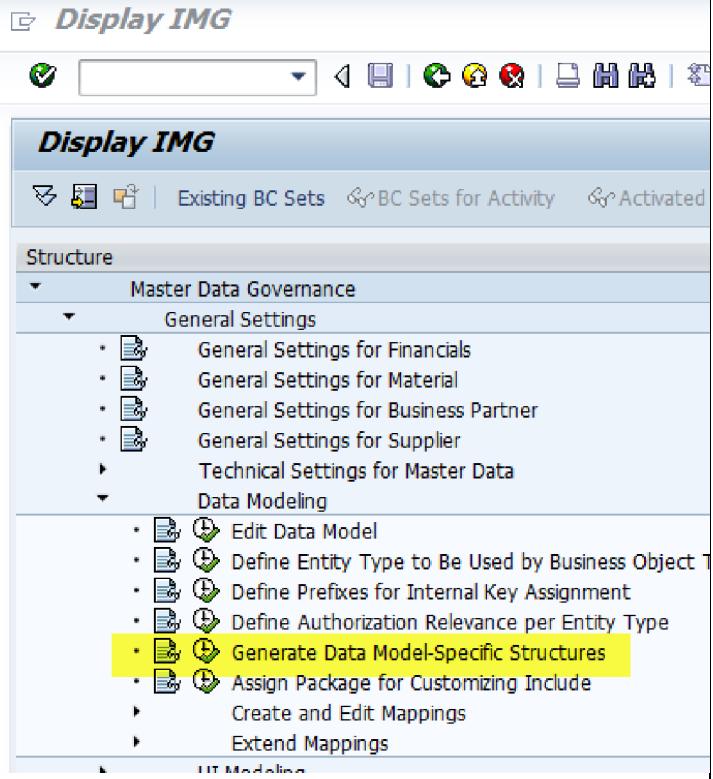
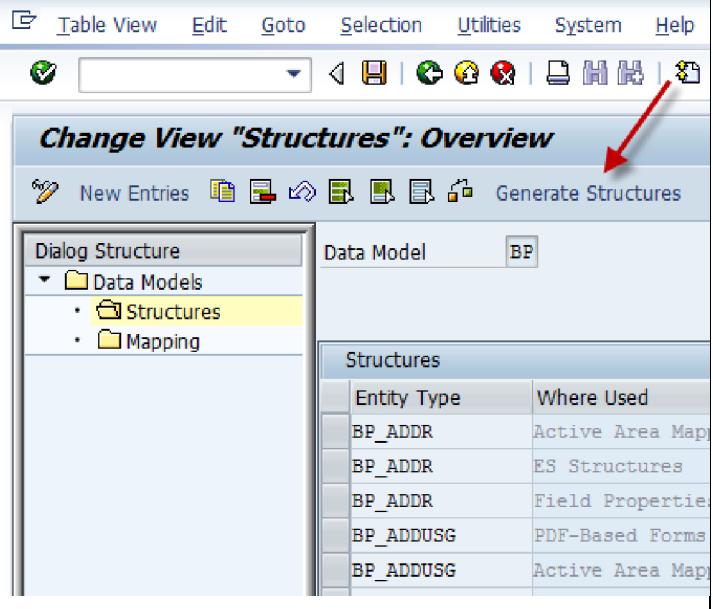
2.2.2. Generate Data Model-Specific Structures for MDG

Whenever you change the MDG data model, you need to regenerate the tables. In this customizing activity, you generate technical structures and tables in the ABAP Dictionary for each data model and entity type. The system uses these structures internally for implementing the staging area. To generate these data modelspecific structures follow the steps below.

Note:

In general if you change a data model (for example, if you change attributes of entity types or relationships), you need to regenerate the structures.

1	Log into system for cross-client maintenances.	
---	------------------------------------------------	--

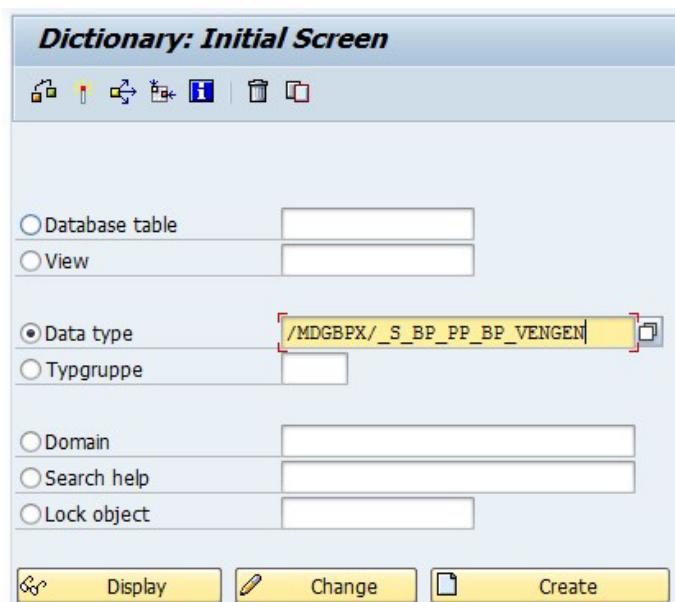
2 Start Customizing for Master Data Governance (transaction MDGIMG). Select General Settings -> Data Modeling -> Generate Data Model-Specific Structures .	 <p>The screenshot shows the SAP Display IMG dialog. In the left pane, under 'Master Data Governance' > 'General Settings' > 'Data Modeling', the 'Generate Data Model-Specific Structures' button is highlighted with a yellow box.</p>																								
3 Select the row with Data Model BP . Double-click Structures in the left hand panel. Choose the Generate Structures pushbutton.	 <p>The screenshot shows the SAP 'Change View "Structures": Overview' dialog. On the right, a table lists structures for Data Model 'BP'. On the left, the 'Dialog Structure' tree has 'Structures' selected. A red arrow points to the 'Generate Structures' button in the toolbar at the top right of the screen.</p> <table border="1" data-bbox="759 1253 1049 1619"> <thead> <tr> <th colspan="2">Dialog Structure</th> </tr> </thead> <tbody> <tr> <td>-</td> <td>Data Models</td> </tr> <tr> <td>-</td> <td> • Structures</td> </tr> <tr> <td>-</td> <td> • Mapping</td> </tr> </tbody> </table> <table border="1" data-bbox="1060 1253 1456 1619"> <thead> <tr> <th colspan="2">Data Model BP</th> </tr> <tr> <th colspan="2">Structures</th> </tr> <tr> <th>Entity Type</th> <th>Where Used</th> </tr> </thead> <tbody> <tr> <td>BP_ADDR</td> <td>Active Area Map</td> </tr> <tr> <td>BP_ADDR</td> <td>ES Structures</td> </tr> <tr> <td>BP_ADDR</td> <td>Field Proprie</td> </tr> <tr> <td>BP_ADDUSG</td> <td>PDF-Based Forms</td> </tr> <tr> <td>BP_ADDUSG</td> <td>Active Area Map</td> </tr> </tbody> </table>	Dialog Structure		-	Data Models	-	• Structures	-	• Mapping	Data Model BP		Structures		Entity Type	Where Used	BP_ADDR	Active Area Map	BP_ADDR	ES Structures	BP_ADDR	Field Proprie	BP_ADDUSG	PDF-Based Forms	BP_ADDUSG	Active Area Map
Dialog Structure																									
-	Data Models																								
-	• Structures																								
-	• Mapping																								
Data Model BP																									
Structures																									
Entity Type	Where Used																								
BP_ADDR	Active Area Map																								
BP_ADDR	ES Structures																								
BP_ADDR	Field Proprie																								
BP_ADDUSG	PDF-Based Forms																								
BP_ADDUSG	Active Area Map																								

- 4 In the following steps you will verify that your field extension was successfully added to the Customer Include of the mapping structure.

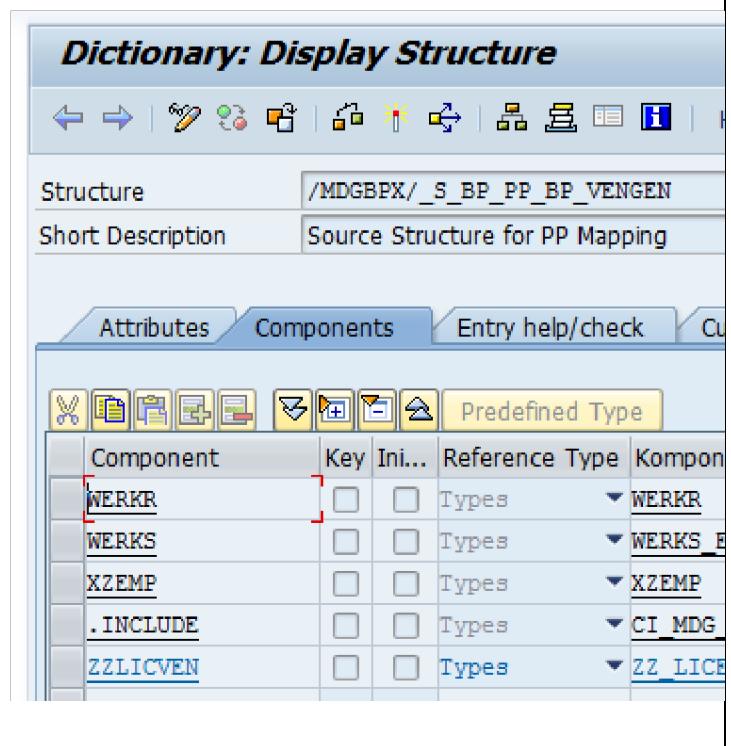
Start transaction **SE11**.

Display structure

/MDGBPX/_S_BP_PP_BP_VENGEN by entering the details as shown.



- 5 The new field has been generated in the Customer Include of the mapping structure.



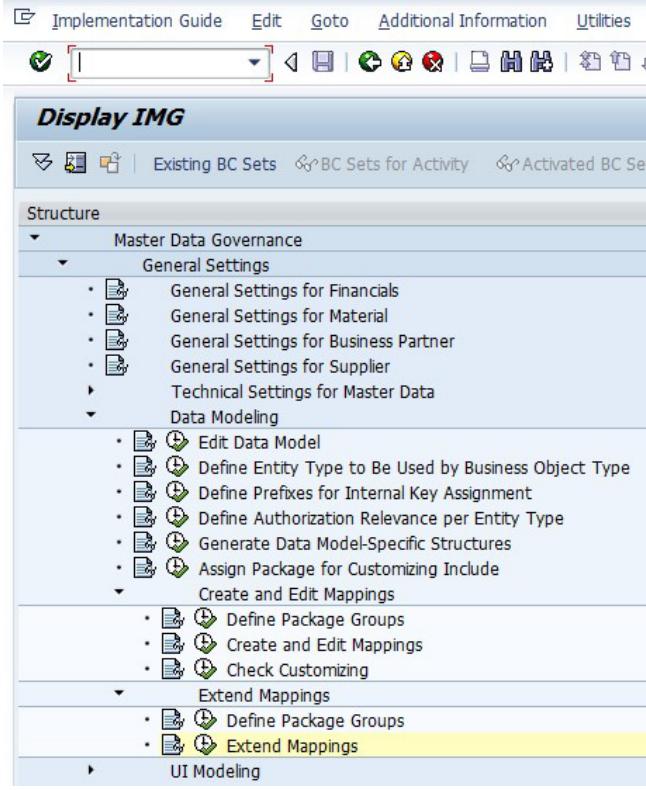
2.2.3. Define SMT Mapping

The Service Mapping Tool (SMT) is a program that enables you to fill target structures by using sets of source structures (in ABAP). It supports simple mappings, mappings with field transformations, complex transformations, and field checks.

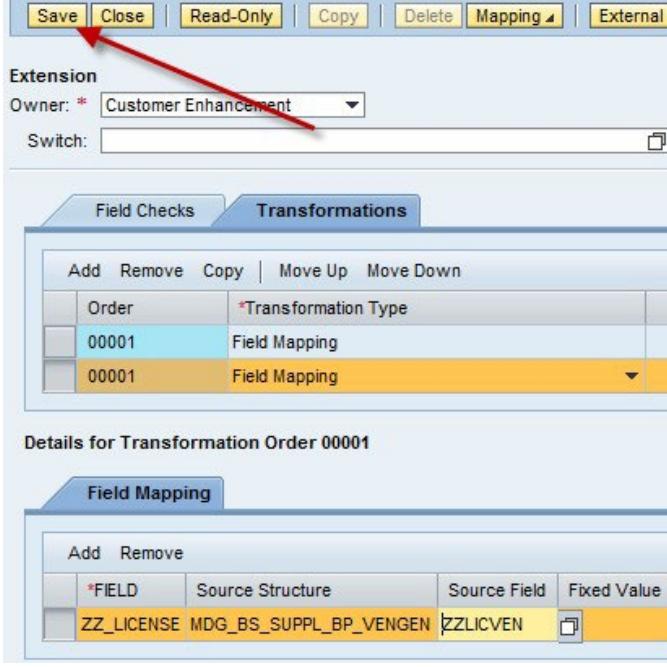
You need to perform the following mappings:

- Mapping Central Data -> API (**SUPPL_BP_VENGEN_2API**)
Central Business Partner Data

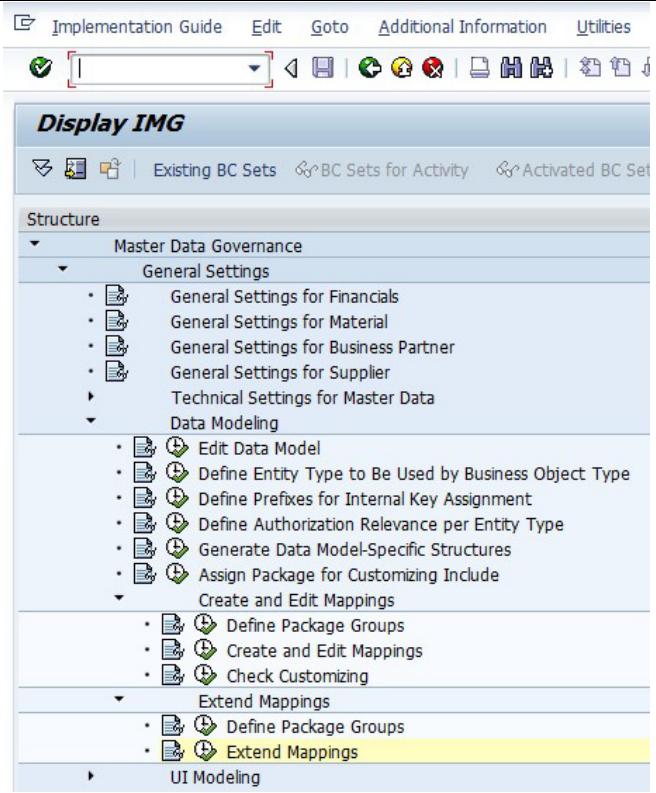
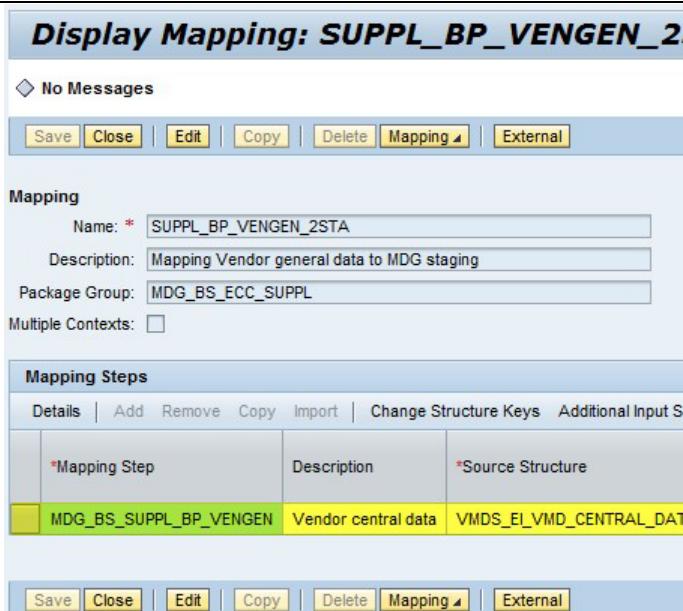
- Mapping Central Data -> Staging(**SUPPL_BP_VENGEN_2STA**)
Central Business Partner Data

1	<p>Log into system for cross-client maintenances.</p>	
2	<p>Start Customizing for Master Data Governance (transaction MDGIMG).</p> <p>Select General Settings -> Data Modeling -> Extend Mappings -> Extend Mappings.</p> <p>Select Mapping -> Open.</p> <p>Enter SUPPL_BP_VENGEN_2API.</p>	 <p>The screenshot shows the SAP Display IMG interface. The title bar includes 'Implementation Guide', 'Edit', 'Goto', 'Additional Information', and 'Utilities'. Below the title bar is a toolbar with various icons. The main area is titled 'Display IMG' and shows a hierarchical navigation structure. The path selected is highlighted in yellow: 'Master Data Governance' > 'General Settings' > 'Data Modeling' > 'Create and Edit Mappings' > 'Extend Mappings'. Other options like 'Edit Data Model' and 'Define Entity Type' are also visible but not selected.</p>

3	<p>Select mapping structure VMDS_EI_VMD_CENTRAL_DATA.</p> <p>Choose the Details pushbutton above the Mapping Steps table.</p>	<p>Display Mapping: SUPPL_BP_VENGEN_2API (Ext)</p> <p>No Messages</p> <p>Save Close Edit Copy Delete Mapping External</p> <p>Mapping</p> <table border="1"> <tr> <td>Name: *</td> <td>SUPPL_BP_VENGEN_2API</td> </tr> <tr> <td>Description:</td> <td>Vendor central data to API</td> </tr> <tr> <td>Package Group:</td> <td>MDG_BS_ECC_SUPPL</td> </tr> <tr> <td>Multiple Contexts:</td> <td><input type="checkbox"/></td> </tr> </table> <p>Mapping Steps</p> <table border="1"> <thead> <tr> <th colspan="3">Details Add Remove Copy Import Change Structure Keys Additional Input Structures Assign</th> </tr> <tr> <th>*Mapping Step</th> <th>Description</th> <th>*Source Structure</th> </tr> </thead> <tbody> <tr> <td>BUS_INSTANCE</td> <td>BP header</td> <td>MDG_BS_SUPPL_BP_V</td> </tr> <tr style="background-color: #ffffcc;"> <td>VMDS_EI_VMD_CENTRAL_DATA</td> <td>Vendor central data</td> <td>MDG_BS_SUPPL_BP_V</td> </tr> <tr> <td>VMDS_EI_VMD_CENTRAL_DATA_XFLAG</td> <td>Update flag for vendor central data</td> <td>MDG_BS_SUPPL_BP_V</td> </tr> <tr> <td>VMDS_INSTANCE</td> <td>Vendor header</td> <td>MDG_BS_SUPPL_BP_V</td> </tr> </tbody> </table> <p>Save Close Edit Copy Delete Mapping External</p>	Name: *	SUPPL_BP_VENGEN_2API	Description:	Vendor central data to API	Package Group:	MDG_BS_ECC_SUPPL	Multiple Contexts:	<input type="checkbox"/>	Details Add Remove Copy Import Change Structure Keys Additional Input Structures Assign			*Mapping Step	Description	*Source Structure	BUS_INSTANCE	BP header	MDG_BS_SUPPL_BP_V	VMDS_EI_VMD_CENTRAL_DATA	Vendor central data	MDG_BS_SUPPL_BP_V	VMDS_EI_VMD_CENTRAL_DATA_XFLAG	Update flag for vendor central data	MDG_BS_SUPPL_BP_V	VMDS_INSTANCE	Vendor header	MDG_BS_SUPPL_BP_V																													
Name: *	SUPPL_BP_VENGEN_2API																																																								
Description:	Vendor central data to API																																																								
Package Group:	MDG_BS_ECC_SUPPL																																																								
Multiple Contexts:	<input type="checkbox"/>																																																								
Details Add Remove Copy Import Change Structure Keys Additional Input Structures Assign																																																									
*Mapping Step	Description	*Source Structure																																																							
BUS_INSTANCE	BP header	MDG_BS_SUPPL_BP_V																																																							
VMDS_EI_VMD_CENTRAL_DATA	Vendor central data	MDG_BS_SUPPL_BP_V																																																							
VMDS_EI_VMD_CENTRAL_DATA_XFLAG	Update flag for vendor central data	MDG_BS_SUPPL_BP_V																																																							
VMDS_INSTANCE	Vendor header	MDG_BS_SUPPL_BP_V																																																							
4	<p>Select the Transformations tab.</p> <p>Change to edit mode by clicking the Edit pushbutton.</p> <p>Above the transformations table choose the Add pushbutton to add a new transformation of type Field Mapping.</p>	<p>Display Mapping Step: VMDS_EI_VMD_CENTRAL_DATA</p> <p>Mapping SUPPL_BP_VENGEN_2API > Mapping Step VMDS_EI_VMD_CENTRAL_DATA</p> <p>Mapping Description Vendor central data to API Package Group MDG_BS_ECC_SU</p> <p>No Messages</p> <p>Save Close Edit Copy Delete Mapping External</p> <p>Extension</p> <p>Owner: * Customer Enhancement</p> <p>Switch:</p> <p>Transformations</p> <table border="1"> <tr> <td>Add</td> <td>Remove</td> <td>Copy</td> <td>Move Up</td> <td>Move Down</td> </tr> <tr> <td></td> <td>Order</td> <td>*Transformation Type</td> <td colspan="2">Conditional</td> </tr> <tr> <td></td> <td>00001</td> <td>Field Mapping</td> <td colspan="2"><input type="checkbox"/></td> </tr> </table> <p>Details for Transformation Order 00001</p> <p>Field Mapping</p> <table border="1"> <tr> <td>Add</td> <td>Remove</td> </tr> <tr> <td></td> <td></td> </tr> <tr> <td>*FIELD</td> <td>Source Structure</td> <td>Source Field</td> <td>Fixed Value</td> <td>Conversion Class</td> <td>Refere Type</td> </tr> <tr> <td>ACTSS</td> <td>MDG_BS_SUPPL_BP_VENGEN</td> <td>ACTSS</td> <td></td> <td></td> <td><input type="checkbox"/></td> </tr> <tr> <td>BAHNS</td> <td>MDG_BS_SUPPL_BP_VENGEN</td> <td>BAHNS</td> <td></td> <td></td> <td><input type="checkbox"/></td> </tr> <tr> <td>BEGRU</td> <td>MDG_BS_SUPPL_BP_VENGEN</td> <td>REF_VENGE</td> <td></td> <td></td> <td><input type="checkbox"/></td> </tr> <tr> <td>DLGRP</td> <td>MDG_BS_SUPPL_BP_VENGEN</td> <td>DLGRP</td> <td></td> <td></td> <td><input type="checkbox"/></td> </tr> <tr> <td>DTAMS</td> <td>MDG_BS_SUPPL_BP_VENGEN</td> <td>DTAMS</td> <td></td> <td></td> <td><input type="checkbox"/></td> </tr> </table>	Add	Remove	Copy	Move Up	Move Down		Order	*Transformation Type	Conditional			00001	Field Mapping	<input type="checkbox"/>		Add	Remove			*FIELD	Source Structure	Source Field	Fixed Value	Conversion Class	Refere Type	ACTSS	MDG_BS_SUPPL_BP_VENGEN	ACTSS			<input type="checkbox"/>	BAHNS	MDG_BS_SUPPL_BP_VENGEN	BAHNS			<input type="checkbox"/>	BEGRU	MDG_BS_SUPPL_BP_VENGEN	REF_VENGE			<input type="checkbox"/>	DLGRP	MDG_BS_SUPPL_BP_VENGEN	DLGRP			<input type="checkbox"/>	DTAMS	MDG_BS_SUPPL_BP_VENGEN	DTAMS			<input type="checkbox"/>
Add	Remove	Copy	Move Up	Move Down																																																					
	Order	*Transformation Type	Conditional																																																						
	00001	Field Mapping	<input type="checkbox"/>																																																						
Add	Remove																																																								
*FIELD	Source Structure	Source Field	Fixed Value	Conversion Class	Refere Type																																																				
ACTSS	MDG_BS_SUPPL_BP_VENGEN	ACTSS			<input type="checkbox"/>																																																				
BAHNS	MDG_BS_SUPPL_BP_VENGEN	BAHNS			<input type="checkbox"/>																																																				
BEGRU	MDG_BS_SUPPL_BP_VENGEN	REF_VENGE			<input type="checkbox"/>																																																				
DLGRP	MDG_BS_SUPPL_BP_VENGEN	DLGRP			<input type="checkbox"/>																																																				
DTAMS	MDG_BS_SUPPL_BP_VENGEN	DTAMS			<input type="checkbox"/>																																																				

5	Choose the Save pushbutton.	 <p>The screenshot shows the SAP Fiori interface for managing field mappings. At the top, there are several buttons: Save (highlighted with a red arrow), Close, Read-Only, Copy, Delete, Mapping, and External. Below these are sections for 'Extension' (Owner: Customer Enhancement) and 'Field Checks' (with tabs for Field Checks and Transformations). Under 'Transformations', there's a table with rows for Order, Transformation Type, and two entries labeled '00001 Field Mapping'. A section titled 'Details for Transformation Order 00001' contains a 'Field Mapping' table with columns for *FIELD, Source Structure, Source Field, and Fixed Value. The first row shows ZZ_LICENSE, MDG_BS_SUPPL_BP_VENGEN, ZZLICVEN, and a checkbox.</p>
6	Select Mapping SUPPL_BP_VENGEN_2API.	<h3>Edit Mapping Step: VMDS_EI_VML</h3> <p>Mapping SUPPL_BP_VENGEN_2API > Mapping Step VMDS_EI_VMD_CENTI</p> <p>Mapping Description Vendor central data to API Package Group</p> <p><input checked="" type="checkbox"/> Mapping SUPPL_BP_VENGEN_2API saved successfully under request E\</p>

7	<p>Select the mapping step Update flag for vendor central data.</p>	<table border="1"> <thead> <tr> <th>*Mapping Step</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>BUS_EI_INSTANCE</td> <td>BP header</td> </tr> <tr> <td>VMDS_EI_VMD_CENTRAL_DATA</td> <td>Vendor central data</td> </tr> <tr style="background-color: #FFD700;"> <td>VMDS_EI_VMD_CENTRAL_DATA_XFLAG</td> <td>Update flag for vendor central data</td> </tr> <tr> <td>VMDS_INSTANCE</td> <td>Vendor header</td> </tr> </tbody> </table>	*Mapping Step	Description	BUS_EI_INSTANCE	BP header	VMDS_EI_VMD_CENTRAL_DATA	Vendor central data	VMDS_EI_VMD_CENTRAL_DATA_XFLAG	Update flag for vendor central data	VMDS_INSTANCE	Vendor header																														
*Mapping Step	Description																																									
BUS_EI_INSTANCE	BP header																																									
VMDS_EI_VMD_CENTRAL_DATA	Vendor central data																																									
VMDS_EI_VMD_CENTRAL_DATA_XFLAG	Update flag for vendor central data																																									
VMDS_INSTANCE	Vendor header																																									
8	<p>Choose the Save pushbutton.</p>	<table border="1"> <thead> <tr> <th>Add</th> <th>Remove</th> <th>Copy</th> <th>Move Up</th> <th>Move Down</th> </tr> </thead> <tbody> <tr> <td>Order</td> <td>*Transformation Type</td> <td></td> <td></td> <td></td> </tr> <tr> <td>00001</td> <td>Field Mapping</td> <td></td> <td></td> <td></td> </tr> <tr style="background-color: #FFD700;"> <td>00001</td> <td>Field Mapping</td> <td></td> <td></td> <td></td> </tr> </tbody> </table> <p>Details for Transformation Order 00001</p> <table border="1"> <thead> <tr> <th colspan="5">Field Mapping</th> </tr> <tr> <th>Add</th> <th>Remove</th> <th></th> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>*FIELD</td> <td>Source Structure</td> <td>Source Field</td> <td>Fixed Value</td> <td>Conversion C</td> </tr> <tr> <td>ZZ_LICENSE</td> <td>MDG_BS_SUPPL_BP_VENGEN_X</td> <td>ZZLICVEN</td> <td><input type="text"/></td> <td></td> </tr> </tbody> </table> <p><small>Tree structures are available to the user to automatically Assign or Map structure fields to a Field Char</small></p>	Add	Remove	Copy	Move Up	Move Down	Order	*Transformation Type				00001	Field Mapping				00001	Field Mapping				Field Mapping					Add	Remove				*FIELD	Source Structure	Source Field	Fixed Value	Conversion C	ZZ_LICENSE	MDG_BS_SUPPL_BP_VENGEN_X	ZZLICVEN	<input type="text"/>	
Add	Remove	Copy	Move Up	Move Down																																						
Order	*Transformation Type																																									
00001	Field Mapping																																									
00001	Field Mapping																																									
Field Mapping																																										
Add	Remove																																									
*FIELD	Source Structure	Source Field	Fixed Value	Conversion C																																						
ZZ_LICENSE	MDG_BS_SUPPL_BP_VENGEN_X	ZZLICVEN	<input type="text"/>																																							

9	<p>Start Customizing for Master Data Governance (transaction MDGIMG).</p> <p>Select General Settings -> Data Modeling -> Extend Mappings -> Extend Mappings.</p> <p>Select Mapping -> Open.</p> <p>Enter SUPPL_BP_VENGEN_2STA.</p>	
10	<p>Select mapping structure MDG_BS_SUPPL_BP_VENGEN.</p> <p>Choose Details above the Mapping Steps table.</p>	

11 Select the **Transformations** tab.

Change to edit mode by clicking the **Edit** pushbutton.

Above the transformations table choose the **Add** pushbutton to add a new transformation of type **Field Mapping**.

2.2.4. Test Your Configuration

To test your configuration you should first add your new field to the MDG single maintenance UI. Section *User Interface Configuration* provides some instructions on how to add the new field to the UI. Create a new change request in MDG and enter a value for the new field in the ERP Vendor Control Data section. Then process the change request until it is activated. Write down the ID of the new Supplier after activation.

Reason:	Standard: <input checked="" type="checkbox"/>	Administrative Data
		License: HD-XX 2222
		ERP Vendor:
		* Account Group: Vendor (int.number assgnmnt)
Authorization Group:	Group Key:	Procurement Monitoring
		Factory Calendar:
Trading Partner:		

Run transaction **SE16** and find your newly created Vendor in table LFA. Make sure the field **ZZ_LICENSE** is there and contains the right value.

Table LFA1 Display

Check Table...

PSOONS	
PSOVN	
PSOTL	
PSOHS	
PSOST	
TRANSPORT CHAIN	
STAGING TIME	0
SCHEDULING TYPE	
SUBMI RELEVANT	
ZMASCHMI	
ZSDN CI	
ZZ SECRET ALIAS	
ZZ LICENSE	HD-YY 3333
ZZLICVEN	HD-XX 2222

filled in by

CVI

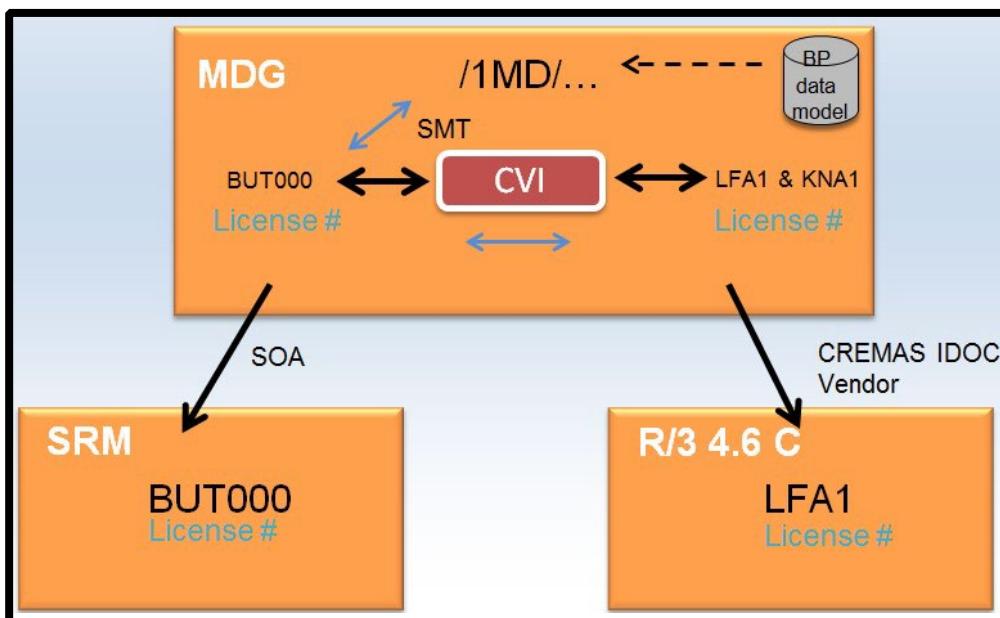
filled by

SMT

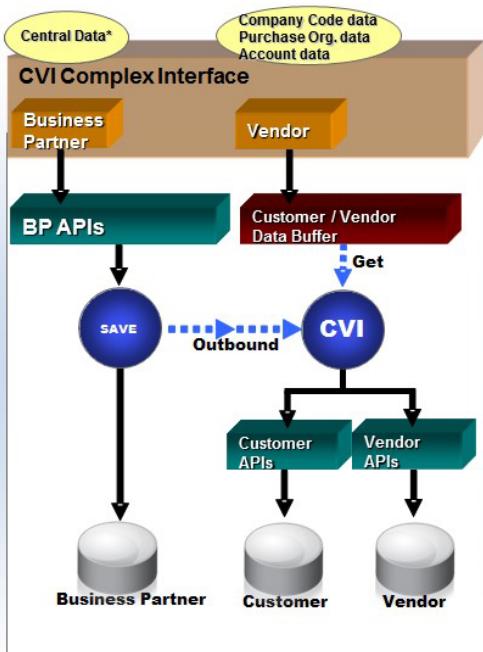
mapping
from MDG

2.3. Extended Field Exists in Business Partner and in Vendor / Customer

In this section you will learn how to add a new field “**License**” to the MDG **BP** data model. In the figure below you can see that it is necessary to extend the MDG **BP** data model. You also must have extended the **BUT000** and **LFA1** / **KNA1** tables on the MDG hub and client systems. In order to facilitate data transfer between the MDG staging area and the **BUT000** on the MDG hub you have to extend the SMT mapping. To facilitate data transfer from **BUT000** on the hub and the **LFA1** / **KNA1** tables on the MDG hub you have to implement CVI mapping. You find more information about Customer Vendor Integration in this FAQ Note [2713963](#) FAQ: CVI - Customer Vendor Integration for system conversion to SAP S/4HANA.



In this scenario the Customer Vendor Integration (CVI) is taking care of the data transfer between **BUT000** and **LFA1** / **KNA1** data structures. For a better understanding you can take a look at the CVI architecture overview shown in the figure below.



This section is subdivided in to one sub-section for each implementation step. Take a look at the table below to get an overview of the steps involved.

Implementation Steps	Field only in Business Partner	Field only in Vendor/Customer	Field in both BP and Vendor/ Customer
Extend ERP data model	✓	✓	✓
Extend MDG data model	✓	✓	✓
Generate MDG data model	✓	✓	✓
Define SMT mapping	✓ ¹	✓ ²	✓ ¹
Implement CVI mapping			✓

SMT Mappings
¹ MDG staging ↔ Business Partner
² MDG staging ↔ Vendor / Customer

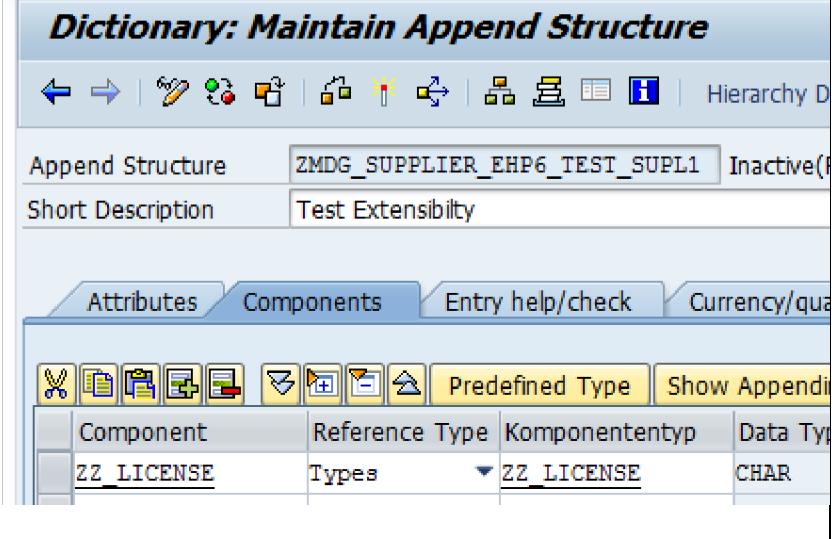
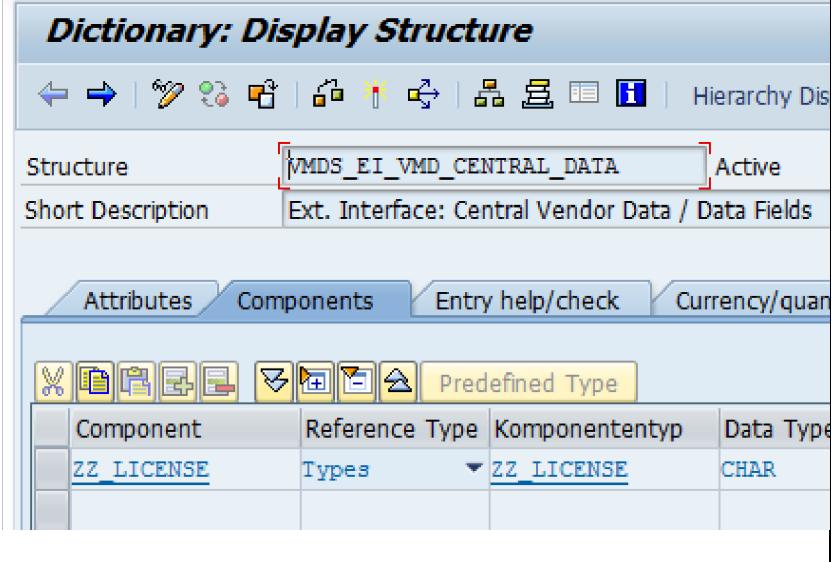
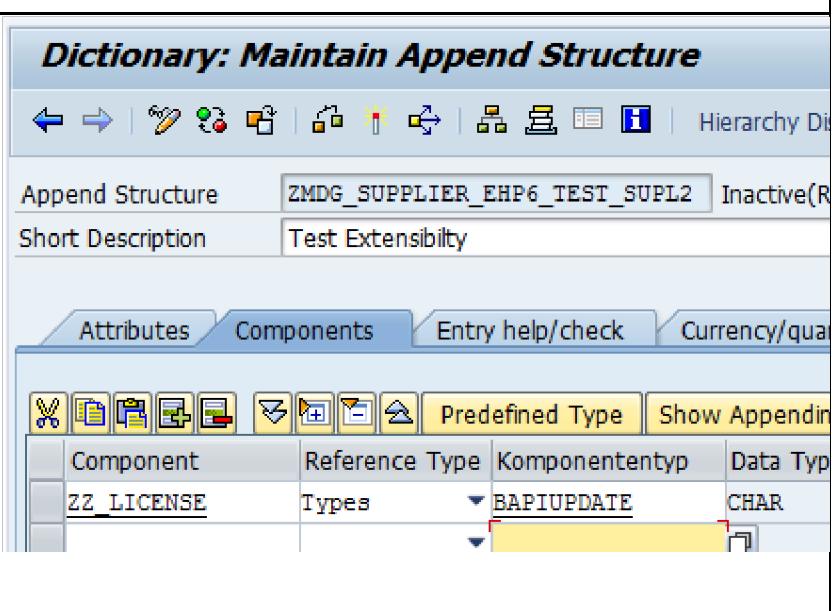
Since this scenario contains several similarities to the one explained in section Extended Field Exists in Business Partner and the one explained in section Extended Field Exists in Vendor / Customer on the detailed instructions below focus only on the additional steps.

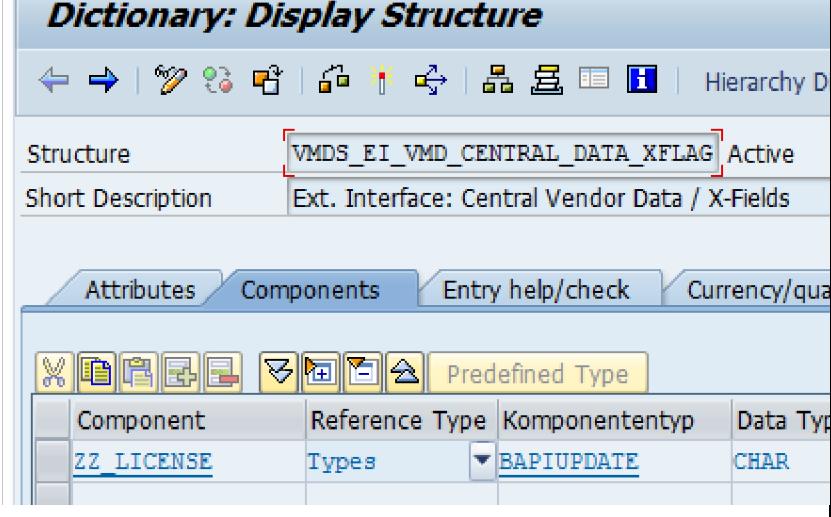
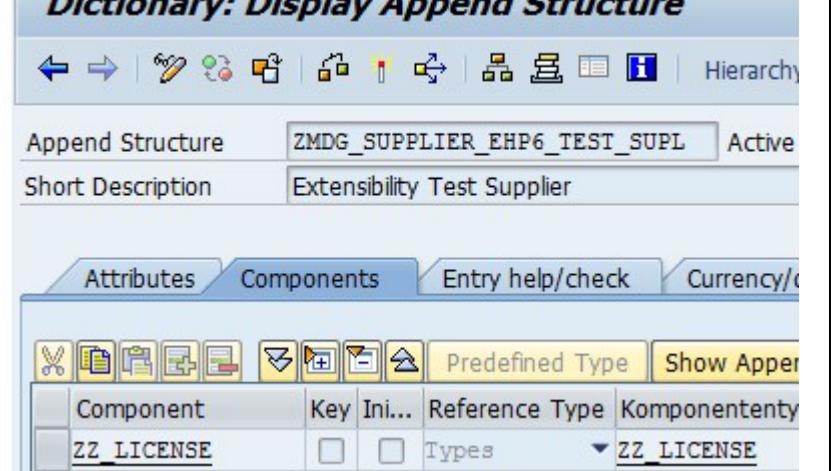
Extend ERP data model	For the extension of BUT000 follow steps in section Extend the BUT000 Data Model starting. To extend LFA1 follow the steps outlined in the next section.
Extend MDG data model	Follow the steps in section <i>Extend the MDG Data Model</i> above
Generate MDG data model specific structures	Follow the steps in section <i>Generate Data Model</i> above.
Define SMT mapping	Follow steps in the section <i>Define SMT Mapping</i> above.
Adjust Staging Area of Linked Change Requests	Follow steps in the section <i>Adjust Staging Area of Linked Change Requests</i> above.
Implement CVI mapping	For implementing the CVI mapping follow the steps below.

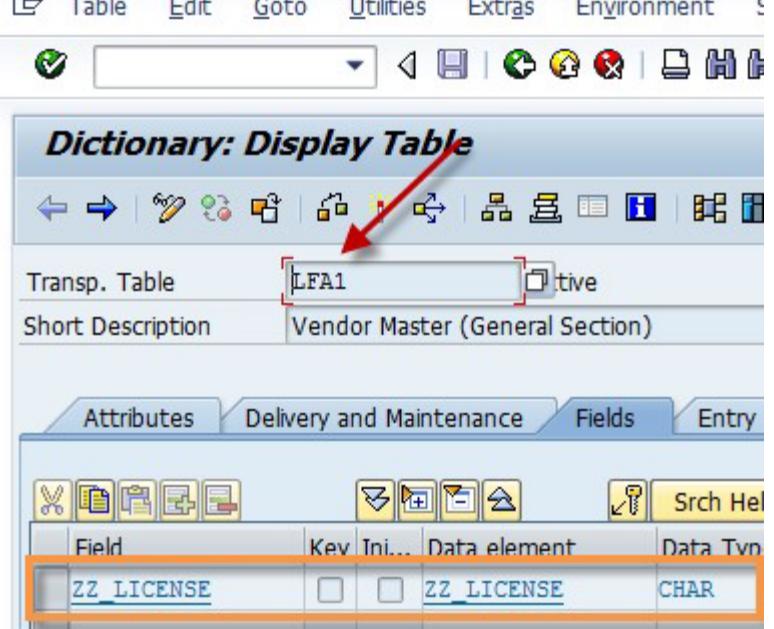
2.3.1. Extend LFA1

In this section you extend the append structures for vendor (**LFA1**). You add the field **ZZ_LICENSE** to structures **LFA1**, **VMDS_EI_VMD_CENTRAL_DATA** and **VMDS_EI_VMD_CENTRAL_DATA_XFLAG**

1	Log into system for cross-client maintenances.	
---	------------------------------------------------	--

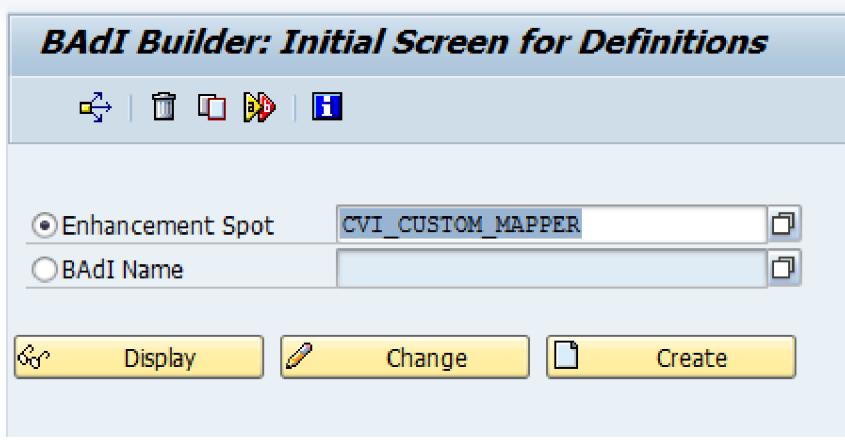
2	<p>Start transaction SE11. Open structure ZMDG_SUPPLIER_EHP6_TEST_S UPL1. Add the new field ZZ_LICENSE to the structure. Activate the changes.</p>	 <p>Dictionary: Maintain Append Structure</p> <p>Append Structure: ZMDG_SUPPLIER_EHP6_TEST_SUPL1 Inactive(R)</p> <p>Short Description: Test Extensibility</p> <table border="1"> <thead> <tr> <th>Component</th> <th>Reference Type</th> <th>Komponententyp</th> <th>Data Type</th> </tr> </thead> <tbody> <tr> <td>ZZ_LICENSE</td> <td>Types</td> <td>ZZ_LICENSE</td> <td>CHAR</td> </tr> </tbody> </table>	Component	Reference Type	Komponententyp	Data Type	ZZ_LICENSE	Types	ZZ_LICENSE	CHAR
Component	Reference Type	Komponententyp	Data Type							
ZZ_LICENSE	Types	ZZ_LICENSE	CHAR							
3	<p>Verify that the appended field appears in structure VMDS_EI_VMD_CENTRAL_DATA.</p>	 <p>Dictionary: Display Structure</p> <p>Structure: VMDS_EI_VMD_CENTRAL_DATA Active</p> <p>Short Description: Ext. Interface: Central Vendor Data / Data Fields</p> <table border="1"> <thead> <tr> <th>Component</th> <th>Reference Type</th> <th>Komponententyp</th> <th>Data Type</th> </tr> </thead> <tbody> <tr> <td>ZZ_LICENSE</td> <td>Types</td> <td>ZZ_LICENSE</td> <td>CHAR</td> </tr> </tbody> </table>	Component	Reference Type	Komponententyp	Data Type	ZZ_LICENSE	Types	ZZ_LICENSE	CHAR
Component	Reference Type	Komponententyp	Data Type							
ZZ_LICENSE	Types	ZZ_LICENSE	CHAR							
4	<p>Start transaction SE11. Open structure ZMDG_SUPPLIER_EHP6_TEST_S UPL2. Add the new field ZZ_LICENSE to the structure. Activate the changes.</p>	 <p>Dictionary: Maintain Append Structure</p> <p>Append Structure: ZMDG_SUPPLIER_EHP6_TEST_SUPL2 Inactive(R)</p> <p>Short Description: Test Extensibility</p> <table border="1"> <thead> <tr> <th>Component</th> <th>Reference Type</th> <th>Komponententyp</th> <th>Data Type</th> </tr> </thead> <tbody> <tr> <td>ZZ_LICENSE</td> <td>Types</td> <td>BAPIUPDATE</td> <td>CHAR</td> </tr> </tbody> </table>	Component	Reference Type	Komponententyp	Data Type	ZZ_LICENSE	Types	BAPIUPDATE	CHAR
Component	Reference Type	Komponententyp	Data Type							
ZZ_LICENSE	Types	BAPIUPDATE	CHAR							

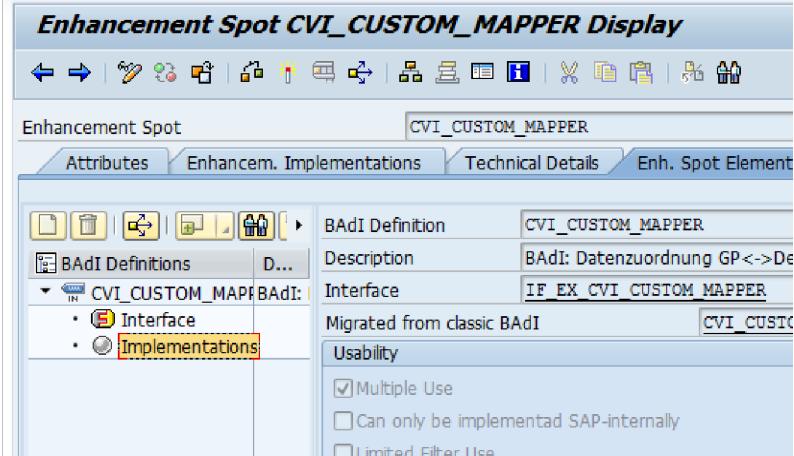
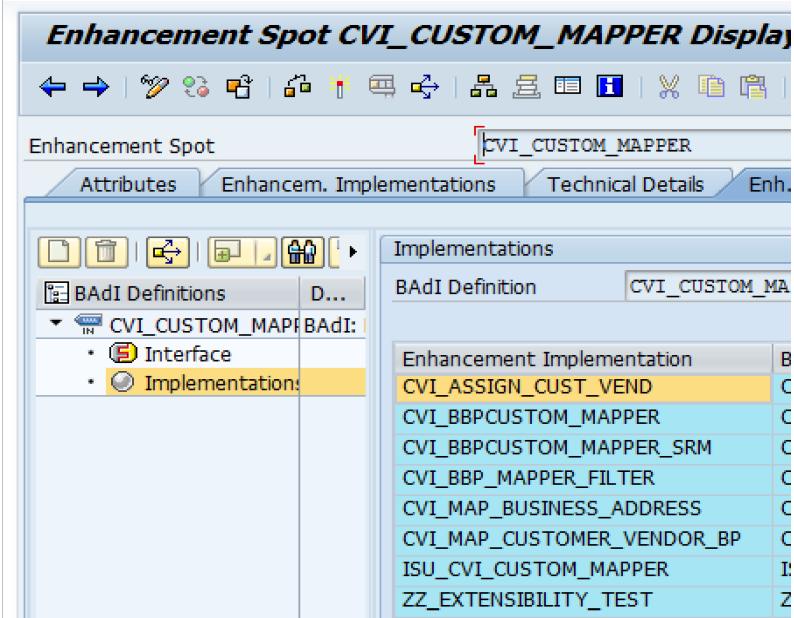
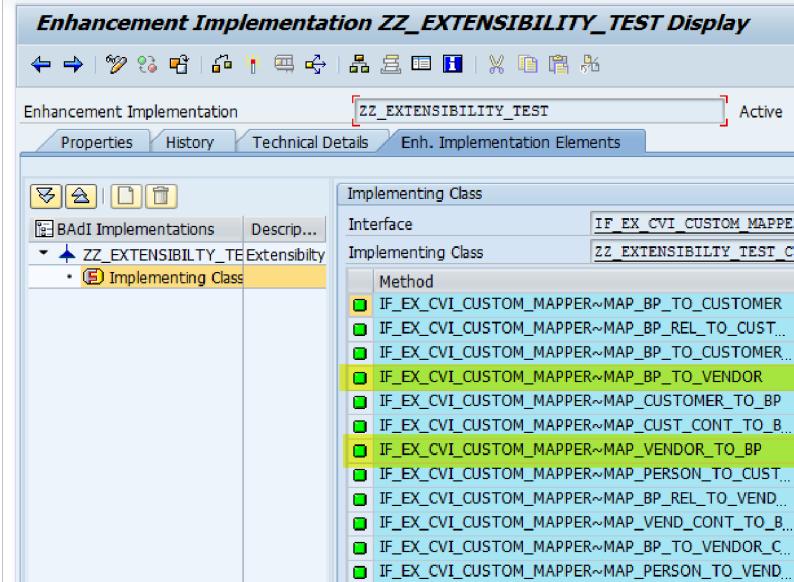
5 Verify that the appended field appears in structure VMDS_EI_VMD_CENTRAL_DATA_XFLAG .	<p>Dictionary: Display Structure</p>  <table border="1"> <thead> <tr> <th>Component</th><th>Reference Type</th><th>Komponententyp</th><th>Data Type</th></tr> </thead> <tbody> <tr> <td>ZZ_LICENSE</td><td>Types</td><td>BAPIUPDATE</td><td>CHAR</td></tr> </tbody> </table>	Component	Reference Type	Komponententyp	Data Type	ZZ_LICENSE	Types	BAPIUPDATE	CHAR		
Component	Reference Type	Komponententyp	Data Type								
ZZ_LICENSE	Types	BAPIUPDATE	CHAR								
6 Start transaction SE11 . Open structure LFA1 . Create an append structure and add the new field ZZ_LICENSE to the append structure. Activate the changes.	<p>Dictionary: Display Append Structure</p>  <table border="1"> <thead> <tr> <th>Component</th><th>Key</th><th>Ini...</th><th>Reference Type</th><th>Komponententyp</th> </tr> </thead> <tbody> <tr> <td>ZZ_LICENSE</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td>Types</td><td>ZZ_LICENSE</td></tr> </tbody> </table>	Component	Key	Ini...	Reference Type	Komponententyp	ZZ_LICENSE	<input type="checkbox"/>	<input type="checkbox"/>	Types	ZZ_LICENSE
Component	Key	Ini...	Reference Type	Komponententyp							
ZZ_LICENSE	<input type="checkbox"/>	<input type="checkbox"/>	Types	ZZ_LICENSE							

7 Verify that the appended field appears in structure LFA1.	 <p>The screenshot shows the SAP Dictionary: Display Table interface. In the top navigation bar, the 'Table' tab is selected. Below it, the 'Transp. Table' dropdown is set to 'LFA1'. The 'Short Description' is listed as 'Vendor Master (General Section)'. The main area displays a table with four columns: Field, Key, Initial Value, Data element, and Data Type. A row for the field 'ZZ_LICENSE' is highlighted with an orange border. The 'Data element' column shows 'ZZ_LICENSE' and the 'Data Type' column shows 'CHAR'. A red arrow points to the 'LFA1' entry in the 'Transp. Table' dropdown.</p>
-------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.3.2. *Implement Enhancement Spot CVI_CUSTOM_MAPPER*

To map our custom field from business partner to vendor and back you must implement interface **IF_EX_CVI_CUSTOM_MAPPER** of enhancement spot **CVI_CUSTOM_MAPPER**. You should at least implement the methods highlighted in the figure below.

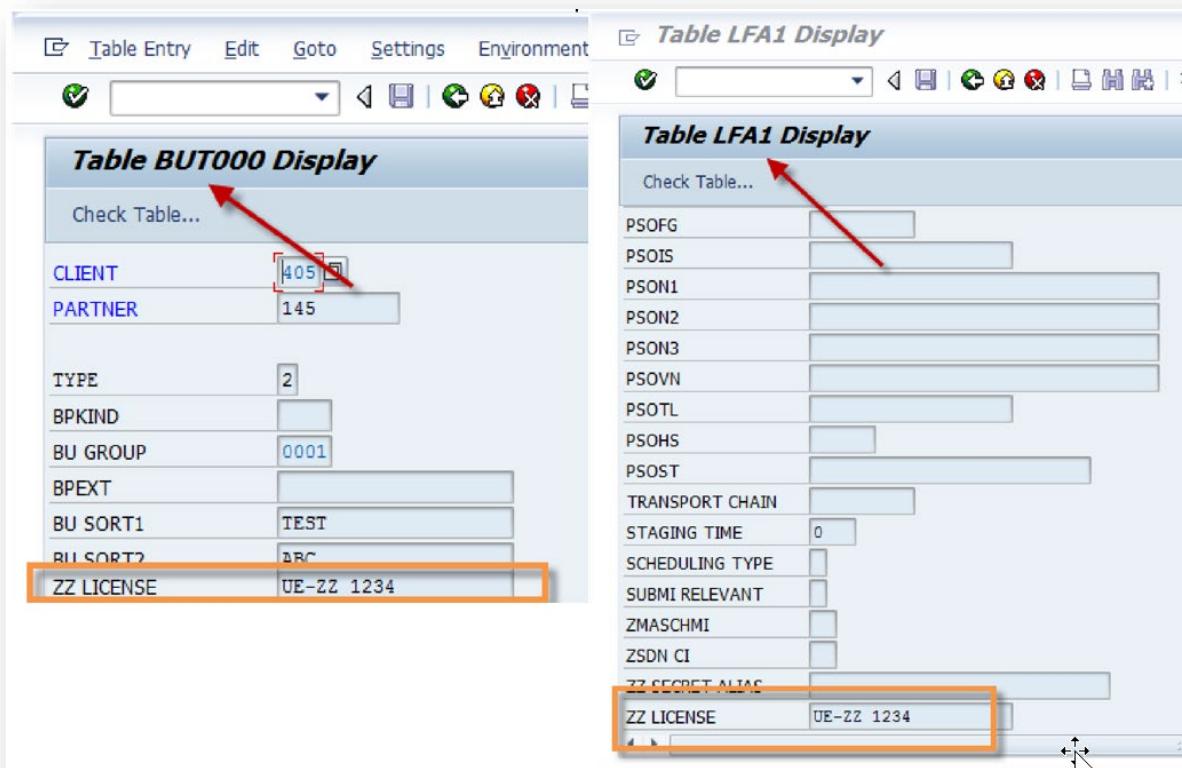
1 Log into system for cross-client maintenances.	
2 Start transaction SE18 . Enter CVI_CUSTOM_MAPPER and choose the Display pushbutton.	 <p>The screenshot shows the BAdI Builder: Initial Screen for Definitions. It has two radio button options: 'Enhancement Spot' (selected) and 'BAdI Name'. The 'Enhancement Spot' input field contains 'CVI_CUSTOM_MAPPER'. At the bottom, there are three buttons: 'Display' (highlighted in yellow), 'Change', and 'Create'.</p>

3	<p>Double-click Implementations.</p>	
4	<p>Create a new BAdI implementation ZZ_EXTENSIBILITY_TEST</p>	
5	<p>You will need to implement the interface methods MAP_BP_TO_VENDOR and MAP_VENDOR_TO_BP.</p>	

6	MAP_BP_TO_VENDOR	<pre>c_vendor-central_data-central-data-zz_license = i_partner-central_data-common-data-zz_license.</pre> <pre>c_vendor-central_data-central-dataxx- zz_license = i_partner-central_data-common- dataxxzz_license.</pre>
7	MAP_VENDOR_TO_BP	<pre>c_partner-central_data-common-data- zz_license = i_vendor-central_data-central- data-zz_license.</pre> <pre>c_partner-central_data-common-dataxx- zz_license = i_vendor-central_data-central-dataxxzz_license.</pre>

2.3.3. Test Your Configuration

To test your configuration you should first add your new field to the MDG single maintenance UI. At the end of this document you will find a section that provides some instructions on how to add the new field to the UI. Create a new change request in MDG and enter a value for the new field. Then process the change request until it is activated. Write down the ID of the new Supplier after activation. Check **BUT000** and **LFA1** the value for the new License field should appear in both tables.



2.4. User Interface Configuration

The UI is configured using the Floorplan Manager. The Floorplan Manager (FPM) is a Web Dynpro ABAP application that provides a framework for developing new Web Dynpro ABAP application interfaces consistent with SAP UI guidelines.

The entry point you need for starting an application is the *application configuration*, which is tied to a single Web Dynpro application. The necessary information needed to start the application is divided between the following two entities:

- **Web Dynpro ABAP Application:** Contains the information about the main component and window of the application
- **Web Dynpro ABAP Application Configuration:** Contains the information about the configuration used for starting the main component

There are only 3 different main components used in FPM-based applications. Each one corresponds to one of the supported floorplans:

- **OIF (Object Instance Floorplan):** component **FPM_OIF_COMPONENT**
- **GAF (Guided Activity Floorplan):** component **FPM_GAF_COMPONENT**
- **OVP (Overview Page Floorplan):** component **FPM_OVP_COMPONENT**

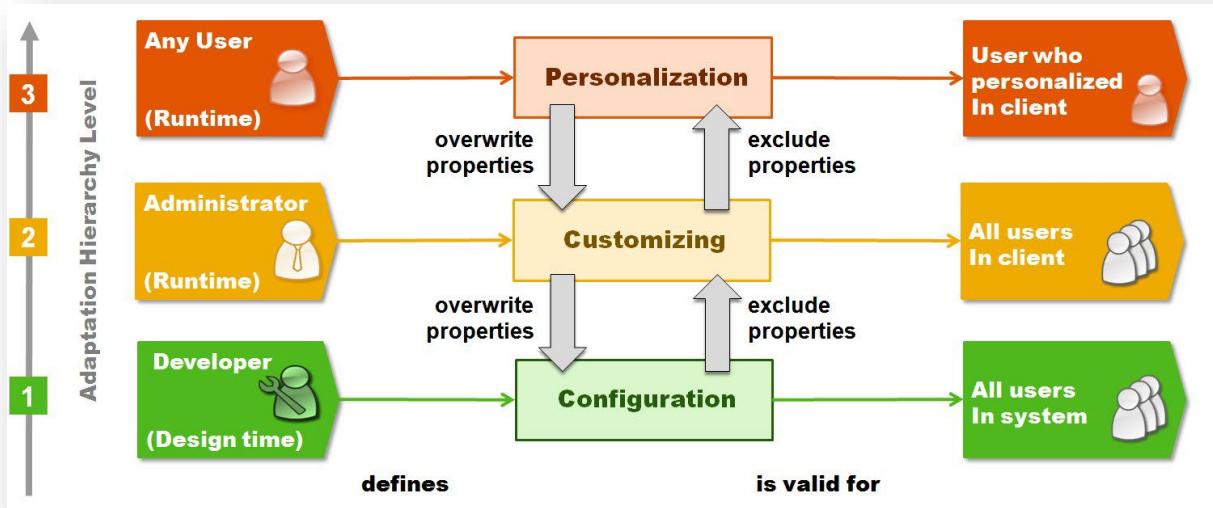
In the screenshot below application BS_OVP_SP is using component **FPM_OVP_COMPONENT**. Here, it is specified that **FPM_OVP_COMPONENT** will start with component configuration BS_SP_OVP. As component **FPM_OVP_COMPONENT** is the component providing the floorplan's functionality and layout, we will use the term 'floorplan component' for it and the term 'floorplan configuration' for the configuration used to start it.

The screenshot shows the 'Application Configuration BS_OVP_SP' dialog. At the top, there are standard toolbar buttons: Save, Cancel, Edit, Check, New Window, Enhance, Properties, and Test. Below the toolbar is a table titled 'Assign Web Dynpro Component'. The table has four columns: 'Component Usage', 'Component', 'Implementation', and 'Configuration Name'. There are two rows. The first row, under 'Component Usage' 'BS_OVP_BP', contains 'FPM_ADAPTABLE_OVP' in the 'Component' column and 'FPM_OVP_COMPONENT' in the 'Implementation' column. The 'Configuration Name' column is empty. The second row, under 'Component Usage' 'OVP', contains 'FPM_OVP_COMPONENT' in both the 'Component' and 'Implementation' columns. The 'Configuration Name' column contains 'BS_SP_OVP'. The entire table is highlighted with a light gray background.

Assign Web Dynpro Component			
Assign Configuration Name			
Component Usage	Component	Implementation	Configuration Name
▼ BS_OVP_BP	FPM_ADAPTABLE_OVP	FPM_ADAPTABLE_OVP	BS_OVP_BP_CBA
▪ OVP	FPM_OVP_COMPONENT	FPM_OVP_COMPONENT	BS_SP_OVP

2.4.1. Adaptation Options in Floorplan Manager

A Floorplan Manager UI can be adapted using different techniques. The figure below shows how the options configuration, customizing and personalization are related to each other. Context-Based-Adaptation is another way the user interface can be customized for specific use cases.



In the context of MDG you would typically choose to *customize* the SAP delivered configuration. Only if customizing is not feasible you would copy the SAP delivered UI configuration to the customer namespace and change the copy.

In the following cases the UI should be copied rather than customized:

- Code changes are required
- The UI needs to be changed for all users in the system and not only client specific
- The changes to the UI are extensive

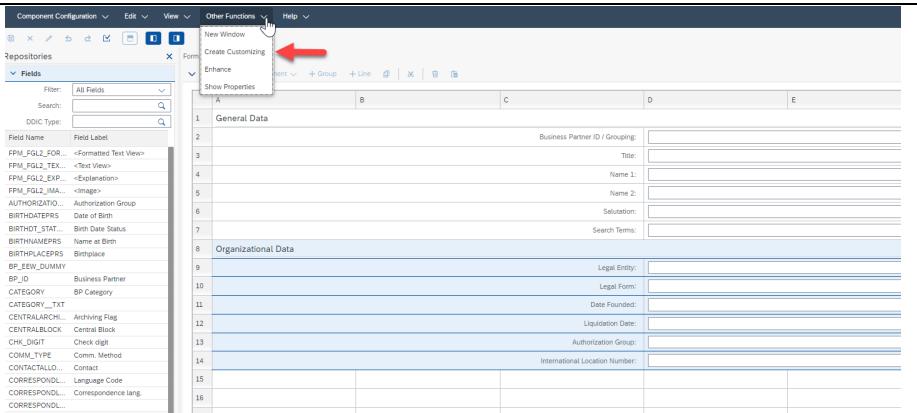
Note:

For more details regarding options for floorplan manager user interface adaptation, advantages, disadvantages and steps required please familiarize yourself with SAP Note [1619534](#) and [Blog](#) and SAP Online Help ("Adapting FPM Applications" <https://help.sap.com/viewer/9c91640bf34c49d8bad6bc560bedd707/latest/en-US/a895eb30653c4a82b675d9258f9ca4d5.html>).

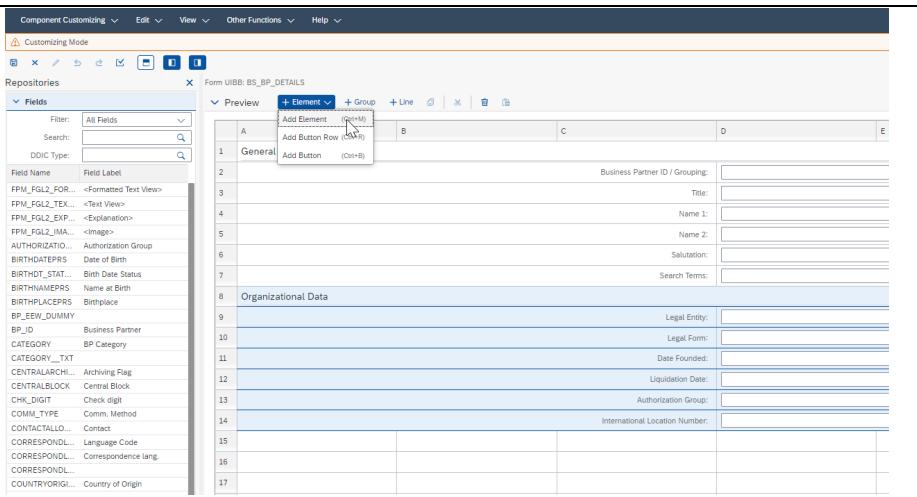
2.4.2. Adding a New Field to UI for Single Maintenance (BUT000 Extension Case)

At this stage you have to make a decision whether you want to extend the standard UI configuration or create a copy. If you are only going to make small changes to the UI configuration (for example adding a field) you can follow the steps described below to extend the standard configuration. In case you need to make significant adjustments to the UI configuration it is recommended to modify a copy.

1 Open application configuration BS_OVP_SP or component configuration BS_BP_DETAILS (or the corresponding configurations for MDG-C/S) and create Customizing. Enter a description and assign a transport request.

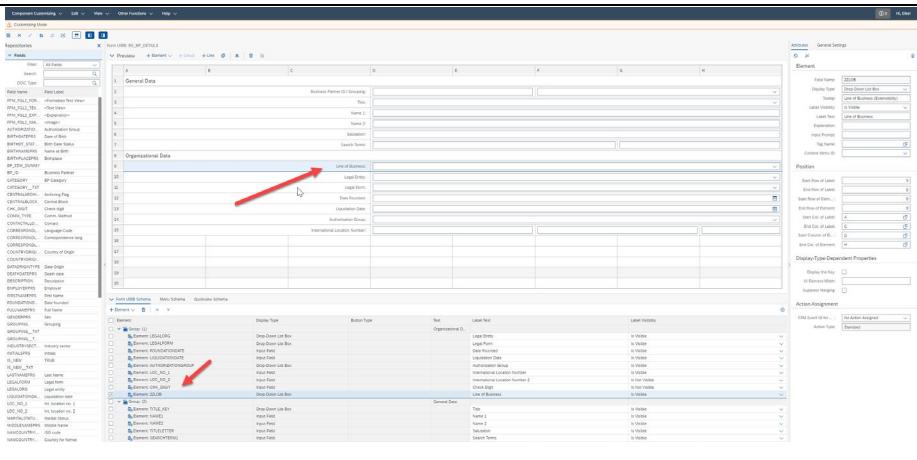


2 You can now continue to make the necessary adjustments. Add field ZZLOB as element.



3 Maintain the attributes for the new field.

Save your component customizing.



4 Start the UI for Supplier Governance and check your new field.

The screenshot shows the SAP Fiori Change Request application interface. At the top, it displays 'Organization: \$189, (no description available)'. Below this, there are tabs for 'Change Request' and 'Supplier Details'. The 'Supplier Details' tab is active, showing various input fields for business partner information. One specific field, 'Line of Business', is highlighted with a yellow background, indicating it is the new field being checked. Other visible fields include 'Business Partner ID / Grouping', 'Name 1', 'Name 2', 'Salutation', 'Search Terms', 'Legal Entity', 'Legal Form', 'Date Founded', 'Liquidation Date', 'Authorization Group', 'International Location Number', and various address and communication details like 'Street / House Number', 'Postal Code / City', 'Country/Region', 'Region', 'Language', 'Transportation Zone', 'PO Box Postal Code / PO Box', 'PO Box Without Number', 'Company Postal Code', 'Telephone (Country/Region, Number, Ex...)', 'Mobile (Country/Region, Number)', 'Fax (Country/Region, Number, Extension)', 'Email', and 'Website'.

Note:

If you want an overview of the UI Customizing settings or if you need to delete UI Customizing settings, refer to the following Web Dynpro applications: **WD_ANALYZE_CONFIG_USER** (user perspective), **WD_ANALYZE_CONFIG_COMP** (component perspective), and **WD_ANALYZE_CONFIG_APPL** (application perspective).

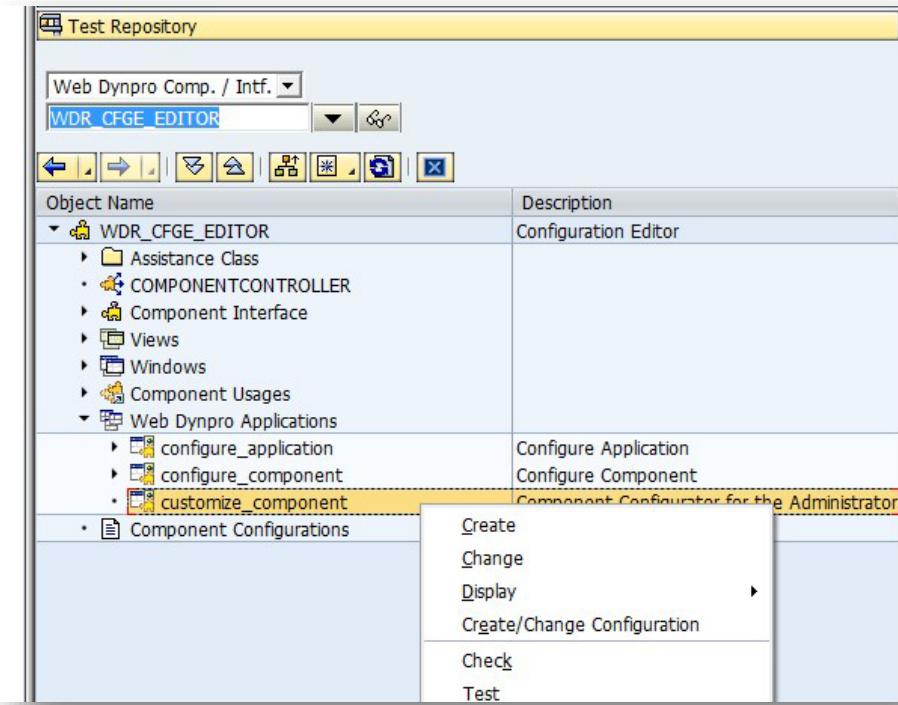
Note:

UI customizing is client specific.

2.4.3. Adding a New Field to the UI for Single Maintenance (Extended Field Exists in Vendor / Customer Only)

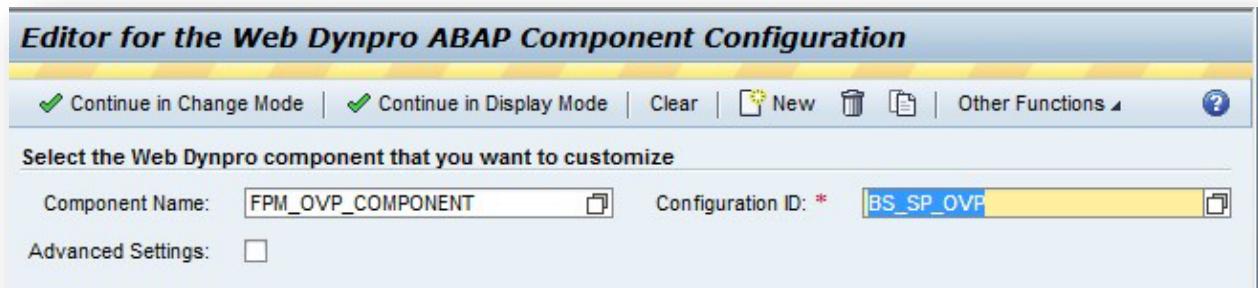
At this stage you have to make a decision whether you want to extend the standard UI configuration or create a copy. If you are only going to make small changes to the UI configuration (for example adding a field) you can follow the steps described below to extend the standard configuration. In case you need to make significant adjustments to the UI configuration it is recommended to modify a copy.

Run transaction **SE80**. And open the **WDR_CFG_Editor Web Dynpro Component**. Start the **customize_component Web Dynpro Application**



To customize the Web Dynpro component, complete the following entries and choose the **New** pushbutton:

- **Component Name:** FPM_OVP_COMPONENT
- **Configuration ID:** BS_SP_OVP



Enter a description and assign a transport request. If you have extended the data model and you want to display additional fields on the user interface, you can now continue to make the necessary adjustments.

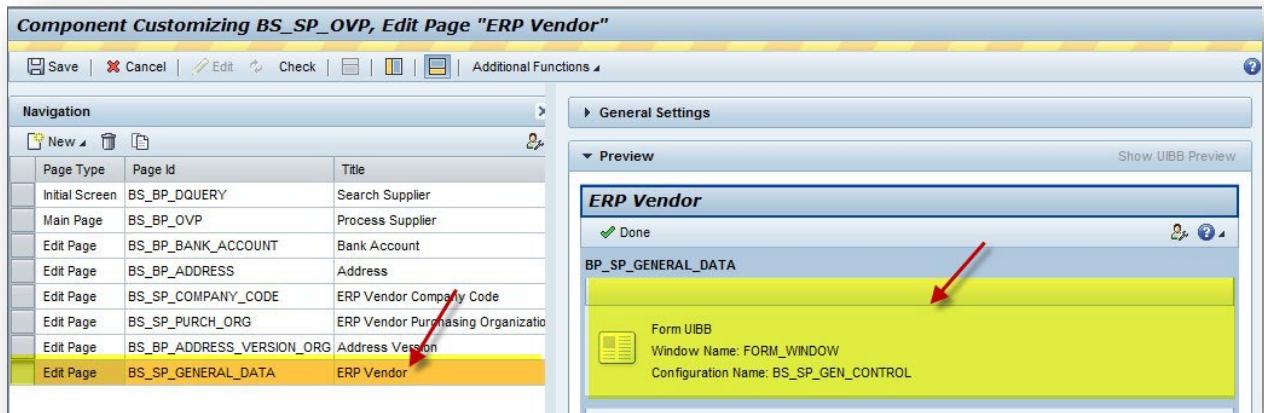


FIGURE: SELECTING THE PAGE BS_SP_GENERAL_DATA (ERP VENDOR) TO ADD THE NEW FIELD

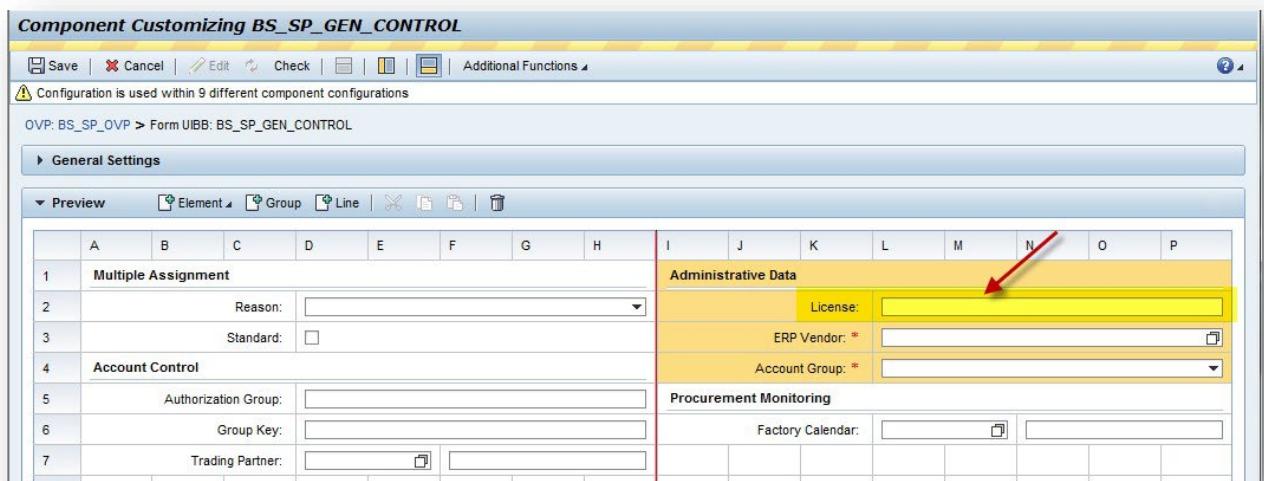


FIGURE: ADDING THE NEW FIELD TO THE BS_SP_GENERAL_DATA (ERP VENDOR) PAGE

3. ADDITIONAL INFORMATION

3.1. Further Reading

3.1.1. **Information on SAP MDG on SAP S/4HANA**

- Exchange knowledge: [SAP Community](#) | [Q&A](#) | [Blog](#)
- Try SAP Master Data Governance on S/4HANA for free: [Trial Version](#)
- Try SAP Master Data Governance on S/4HANA on the SAP Cloud Appliance Library: [S/4HANA 2022 FPS1](#)
- Learn more: [Latest Release](#) | [Help Portal](#) | [How-to Information](#) | [Key Presentations](#)

3.1.2. **SAP Roadmap Explorer**

- Please see the [roadmap for SAP Master Data Governance](#)

3.1.3. **Related Information**

- Learn more: [Floorplan Manager for Web Dynpro ABAP](#) | [How to Adapt FPM](#) | [FPM Blog](#) | [How-to Information](#) | [Service Mapping Tool](#) | [SAP S/4HANA Cookbook CVI](#)

3.2. SAP Notes

In addition to the detailed explanations written in this document, please see the following SAP Notes for further important information about creating a user interface for multiple-record processing:

Note	Description
3372801	Upgrade or Conversion for Master Data Governance, Central Governance
2221398	MDG-BP/C/S/CA: (Un-)Supported Fields in Data Model BP
2847807	MDG-BP/C/S/CA: Usage of MDG Tools and Processes
2313368	Functional restrictions in MDG for Business Partner / Customer / Supplier with SAP Master Data Governance 9.0
2472845	Functional restrictions in MDG for Business Partner / Customer / Supplier with SAP Master Data Governance 9.1
2656712	Functional restrictions in MDG for Business Partner / Customer / Supplier in SAP Master Data Governance 9.2 and on SAP S/4HANA 1809
2816557	Functional restrictions in MDG for Business Partner / Customer / Supplier on SAP S/4HANA 1909
2925030	Functional restrictions in MDG for Business Partner / Customer / Supplier on SAP S/4HANA 2020
3070003	Functional restrictions in MDG for Business Partner / Customer / Supplier on SAP S/4HANA 2021
3220117	Functional restrictions in MDG for Business Partner / Customer / Supplier on SAP S/4HANA 2022
3374711	Functional restrictions in MDG for Business Partner / Customer / Supplier on SAP S/4HANA 2023
3043582	MDG Customer Connection 2020
3194967	MDG Customer Connection 2021 for S/4HANA 2022

<u>3311039</u>	MDG Customer Connection 2023
<u>3428179</u>	Master Data Governance: Continuous Influence
<u>2950412</u>	Functional restrictions in MDG Process Analytics on SAP S/4HANA 2020
<u>3066855</u>	Functional restrictions in MDG Process Analytics on SAP S/4HANA 2021
<u>3225098</u>	Functional restrictions in MDG Process Analytics on SAP S/4HANA 2022
<u>3381795</u>	Functional restrictions in MDG Process Analytics on SAP S/4HANA 2023
<u>2713963</u>	FAQ: CVI - Customer Vendor Integration for system conversion to SAP S/4HANA
<u>3163839</u>	MDG: Issue with Length of Fieldnames in Highlight Changes API
<u>2479869</u>	Usage of Lean Classification with SAP Master Data Governance
<u>1619534</u>	How to Create, Enhance and Adapt FPM Applications
<u>1637249</u>	MDG: Information for efficient message processing
<u>2105467</u>	MDG Performance
<u>2561461</u>	Scope of support for SAP Master Data Governance (MDG)
<u>1637249</u>	MDG: Information for efficient message processing

4. APPENDIX

4.1. Example coding for class ZCL_CUSTOM_BP_HANDLER

Redefinition for IF_MDG_BS_BP_ACCESS_HANDLER~EXECUTE_QUERY.

```
METHOD if_mdg_bs_bp_access_handler~execute_query.  
*! Execute search queries (e.g. triggered by file download).  
  
* IT_SEL can contain multiple and different selection criteria.  
* Still, all criteria relates to the given IV_ENTITY.  
  
* ET_DATA can have a generic structure. It does not need to  
* contain all attributes of the given IV_ENTITY.  
  
DATA:  
    lr_data  TYPE REF TO data,  
    lt_value TYPE usmd_ts_val_list,  
    lv_value TYPE usmd_value.  
  
ef_handled = abap_true.  
  
"The MDG framework executes non-supported queries multiple times  
"(e.g. a second call with less nor no selection criteria). This  
"cannot be handled since it might cause a selection of all existing  
"business partners!  
IF me->mv_prev_query_not_supported = abap_true  
    AND it_sel IS INITIAL  
    AND iv_num_entries = 0.  
    ef_not_supported = me->mv_prev_query_not_supported.  
    RETURN.  
ELSE.  
    CLEAR me->mv_prev_query_not_supported.  
ENDIF.  
  
"Handle the queries for the given entity type  
CLEAR lt_value.  
CASE iv_entity.  
    WHEN if_mdg_bp_constants=>gc_bp_model_entity-central.  
        "The examples relates to attributes of entity type BP_CENTRL.  
        "Since the existing code in class CL_MDG_BS_BP_HANDLER in  
        "method QUERY_BP_CENTRL cannot handle the custom attributes,  
        "the SAP owned class returns a 'ef_not_supported = abap_true'.  
        "Thus the custom handler class is the place to handle that query.  
        "The code of the SAP owned class can used as template. It needs  
        "to be enhanced with the custom attributes.  
    DATA: lt_but000 TYPE TABLE OF but000.  
  
    "Each supported selection criterion needs a range and a line.  
    DATA:  
        lr_auggrp          TYPE RANGE OF but000-auggrp,  
        lr_auggrp_line     LIKE LINE OF lr_auggrp,  
        lr_birthdt         TYPE RANGE OF but000-birthdt,  
        lr_birthdt_line   LIKE LINE OF lr_birthdt,  
        lr_birthdt_status  TYPE RANGE OF but000-birthdt_status,  
        lr_birthdt_status_line  LIKE LINE OF lr_birthdt_status,  
        lr_birthpl         TYPE RANGE OF but000-birthpl,  
        lr_birthpl_line   LIKE LINE OF lr_birthpl,  
        lr_bpext           TYPE RANGE OF but000-bpext,
```

```

lr_bpext_line
lr_bpkind
lr_bpkind_line
lr_bu_langu
lr_bu_langu_line
lr_bu_sort1
lr_bu_sort1_line
lr_bu_sort2
lr_bu_sort2_line
lr_cndsc
lr_cndsc_line
lr_contact
lr_contact_line
lr_deathdt
lr_deathdt_line
lr_emplo
lr_emplo_line
lr_found_dat
lr_found_dat_line
lr_initials
lr_initials_line
lr_jobgr
lr_jobgr_line
lr_langu_corr
lr_langu_corr_line
lr_legal_enty
lr_legal_enty_line
lr_legal_org
lr_legal_org_line
lr_liquid_dat
lr_liquid_dat_line
lr_location_1
lr_location_1_line
lr_location_2
lr_location_2_line
lr_location_3
lr_location_3_line
lr_marst
lr_marst_line
lr_mc_name1
lr_mc_name1_line
lr_mc_name2
lr_mc_name2_line
lr_namcountry
lr_namcountry_line
lr_nameformat
lr_nameformat_line
lr_namemiddle
lr_namemiddle_line
lr_name_first
lr_name_first_line
lr_name_grp1
lr_name_grp1_line
lr_name_grp2
lr_name_grp2_line
lr_name_last
lr_name_last_line
LIKE LINE OF lr_bpext,
TYPE RANGE OF but000-bpkind,
LIKE LINE OF lr_bpkind,
TYPE RANGE OF but000-bu_langu,
LIKE LINE OF lr_bu_langu,
TYPE RANGE OF but000-bu_sort1,
LIKE LINE OF lr_bu_sort1,
TYPE RANGE OF but000-bu_sort2,
LIKE LINE OF lr_bu_sort2,
TYPE RANGE OF but000-cndsc,
LIKE LINE OF lr_cndsc,
TYPE RANGE OF but000-contact,
LIKE LINE OF lr_contact,
TYPE RANGE OF but000-deathdt,
LIKE LINE OF lr_deathdt,
TYPE RANGE OF but000-emplo,
LIKE LINE OF lr_emplo,
TYPE RANGE OF but000-found_dat,
LIKE LINE OF lr_found_dat,
TYPE RANGE OF but000-initials,
LIKE LINE OF lr_initials,
TYPE RANGE OF but000-jobgr,
LIKE LINE OF lr_jobgr,
TYPE RANGE OF but000-langu_corr,
LIKE LINE OF lr_langu_corr,
TYPE RANGE OF but000-legal_enty,
LIKE LINE OF lr_legal_enty,
TYPE RANGE OF but000-legal_org,
LIKE LINE OF lr_legal_org,
TYPE RANGE OF but000-liquid_dat,
LIKE LINE OF lr_liquid_dat,
TYPE RANGE OF but000-location_1,
LIKE LINE OF lr_location_1,
TYPE RANGE OF but000-location_2,
LIKE LINE OF lr_location_2,
TYPE RANGE OF but000-location_3,
LIKE LINE OF lr_location_3,
TYPE RANGE OF but000-marst,
LIKE LINE OF lr_marst,
TYPE RANGE OF but000-mc_name1,
LIKE LINE OF lr_mc_name1,
TYPE RANGE OF but000-mc_name2,
LIKE LINE OF lr_mc_name2,
TYPE RANGE OF but000-namcountry,
LIKE LINE OF lr_namcountry,
TYPE RANGE OF but000-nameformat,
LIKE LINE OF lr_nameformat,
TYPE RANGE OF but000-namemiddle,
LIKE LINE OF lr_namemiddle,
TYPE RANGE OF but000-name_first,
LIKE LINE OF lr_name_first,
TYPE RANGE OF but000-name_grp1,
LIKE LINE OF lr_name_grp1,
TYPE RANGE OF but000-name_grp2,
LIKE LINE OF lr_name_grp2,
TYPE RANGE OF but000-name_last,
LIKE LINE OF lr_name_last,

```

lr_name_last2	TYPE RANGE OF but000-name_last2,
lr_name_last2_line	LIKE LINE OF lr_name_last2,
lr_name_lst2	TYPE RANGE OF but000-name_lst2,
lr_name_lst2_line	LIKE LINE OF lr_name_lst2,
lr_name_org1	TYPE RANGE OF but000-name_org1,
lr_name_org1_line	LIKE LINE OF lr_name_org1,
lr_name_org2	TYPE RANGE OF but000-name_org2,
lr_name_org2_line	LIKE LINE OF lr_name_org2,
lr_name_org3	TYPE RANGE OF but000-name_org3,
lr_name_org3_line	LIKE LINE OF lr_name_org3,
lr_name_org4	TYPE RANGE OF but000-name_org4,
lr_name_org4_line	LIKE LINE OF lr_name_org4,
lr_name1_text	TYPE RANGE OF but000-name1_text,
lr_name1_text_line	LIKE LINE OF lr_name1_text,
lr_natio	TYPE RANGE OF but000-natio,
lr_natio_line	LIKE LINE OF lr_natio,
lr_natpers	TYPE RANGE OF but000-natpers,
lr_natpers_line	LIKE LINE OF lr_natpers,
lrNickname	TYPE RANGE OF but000-nickname,
lrNickname_line	LIKE LINE OF lrNickname,
lr_not_released	TYPE RANGE OF but000-not_released,
lr_not_released_line	LIKE LINE OF lr_not_released,
lr_partgrptyp	TYPE RANGE OF but000-partgrptyp,
lr_partgrptyp_line	LIKE LINE OF lr_partgrptyp,
lr_partner	TYPE RANGE OF but000-partner,
lr_partner_line	LIKE LINE OF lr_partner,
lr_prefix1	TYPE RANGE OF but000-prefix1,
lr_prefix1_line	LIKE LINE OF lr_prefix1,
lr_prefix2	TYPE RANGE OF but000-prefix2,
lr_prefix2_line	LIKE LINE OF lr_prefix2,
lr_print_mode	TYPE RANGE OF but000-print_mode,
lr_print_mode_line	LIKE LINE OF lr_print_mode,
lr_source	TYPE RANGE OF but000-source,
lr_source_line	LIKE LINE OF lr_source,
lr_title	TYPE RANGE OF but000-title,
lr_title_line	LIKE LINE OF lr_title,
lr_title_acal	TYPE RANGE OF but000-title_acal,
lr_title_acal_line	LIKE LINE OF lr_title_acal,
lr_title_aca2	TYPE RANGE OF but000-title_aca2,
lr_title_aca2_line	LIKE LINE OF lr_title_aca2,
lr_title_let	TYPE RANGE OF but000-title_let,
lr_title_let_line	LIKE LINE OF lr_title_let,
lr_title_royl	TYPE RANGE OF but000-title_royl,
lr_title_royl_line	LIKE LINE OF lr_title_royl,
lr_xblk	TYPE RANGE OF but000-xblk,
lr_xblk_line	LIKE LINE OF lr_xblk,
lr_xdete	TYPE RANGE OF but000-xdete,
lr_xdete_line	LIKE LINE OF lr_xdete,
lr_xsexf	TYPE RANGE OF but000-xsexf,
lr_xsexf_line	LIKE LINE OF lr_xsexf,
lr_xsexm	TYPE RANGE OF but000-xsexm,
lr_xsexm_line	LIKE LINE OF lr_xsexm,
lr_zzlob	TYPE RANGE OF but000-
zzlob, "CUSTOM ATTRIBUTE	
lr_zzlob_line	LIKE LINE OF lr_zzlob, "CUSTOM ATTRIBUTE
lr_zz1_zext_wkf1_bus	TYPE RANGE OF but000-
zz1_zext_wkf1_bus, "CUSTOM ATTRIBUTE	

```

        lr_zz1_zext_wkf1_bus_line LIKE LINE OF lr_zz1_zext_wkf1_bus. "CUSTOM ATTRIBUTE
FIELD-SYMBOLS:
<ls_but000> TYPE but000,
<ls_data>   TYPE any,
<ls_sel>    TYPE usmd_s_sel,
<lv_value>  TYPE any.

"map selection to SQL range
LOOP AT it_sel ASSIGNING <ls_sel>.
CASE <ls_sel>-fieldname.
WHEN 'REF_CEN'.
  MOVE-CORRESPONDING <ls_sel> TO lr_augrp_line.
  APPEND lr_augrp_line      TO lr_augrp.
WHEN 'BIRTHDT'.
  MOVE-CORRESPONDING <ls_sel> TO lr_birthdt_line.
  APPEND lr_birthdt_line    TO lr_birthdt.
WHEN 'BIRTHDTST'.
  MOVE-CORRESPONDING <ls_sel> TO lr_birthdt_status_line.
  APPEND lr_birthdt_status_line TO lr_birthdt_status.
WHEN 'BIRTHPL'.
  MOVE-CORRESPONDING <ls_sel> TO lr_birthpl_line.
  APPEND lr_birthpl_line    TO lr_birthpl.
WHEN 'BPEXT'.
  MOVE-CORRESPONDING <ls_sel> TO lr_bpext_line.
  APPEND lr_bpext_line     TO lr_bpext.
WHEN 'BPKIND'.
  MOVE-CORRESPONDING <ls_sel> TO lr_bpkind_line.
  APPEND lr_bpkind_line    TO lr_bpkind.
WHEN 'BU_LANGU'.
  MOVE-CORRESPONDING <ls_sel> TO lr_bu_langu_line.
  APPEND lr_bu_langu_line  TO lr_bu_langu.
WHEN 'BU_SORT1'.
  MOVE-CORRESPONDING <ls_sel> TO lr_bu_sort1_line.
  APPEND lr_bu_sort1_line  TO lr_bu_sort1.
WHEN 'BU_SORT2'.
  MOVE-CORRESPONDING <ls_sel> TO lr_bu_sort2_line.
  APPEND lr_bu_sort2_line  TO lr_bu_sort2.
WHEN 'CNDSC'.
  MOVE-CORRESPONDING <ls_sel> TO lr_cndsc_line.
  APPEND lr_cndsc_line     TO lr_cndsc.
WHEN 'CONTACT'.
  MOVE-CORRESPONDING <ls_sel> TO lr_contact_line.
  APPEND lr_contact_line   TO lr_contact.
WHEN 'DEATHDT'.
  MOVE-CORRESPONDING <ls_sel> TO lr_deathdt_line.
  APPEND lr_deathdt_line   TO lr_deathdt.
WHEN 'EMPLO'.
  MOVE-CORRESPONDING <ls_sel> TO lr_emplo_line.
  APPEND lr_emplo_line     TO lr_emplo.
WHEN 'FOUND_DAT'.
  MOVE-CORRESPONDING <ls_sel> TO lr_found_dat_line.
  APPEND lr_found_dat_line TO lr_found_dat.
WHEN 'INITIALSB'.
  MOVE-CORRESPONDING <ls_sel> TO lr_initials_line.
  APPEND lr_initials_line  TO lr_initials.

```

```

WHEN 'JOBGR'.
  MOVE-CORRESPONDING <ls_sel> TO lr_jobgr_line.
  APPEND lr_jobgr_line      TO lr_jobgr.
WHEN 'LNGU_CORR'.
  MOVE-CORRESPONDING <ls_sel> TO lr_langu_corr_line.
  APPEND lr_langu_corr_line TO lr_langu_corr.
WHEN 'LGAL_ENTY'.
  MOVE-CORRESPONDING <ls_sel> TO lr_legal_enty_line.
  APPEND lr_legal_enty_line TO lr_legal_enty.
WHEN 'LEGAL_ORG'.
  MOVE-CORRESPONDING <ls_sel> TO lr_legal_org_line.
  APPEND lr_legal_org_line TO lr_legal_org.
WHEN 'LQUID_DAT'.
  MOVE-CORRESPONDING <ls_sel> TO lr_liquid_dat_line.
  APPEND lr_liquid_dat_line TO lr_liquid_dat.
WHEN 'LOCATION1'.
  MOVE-CORRESPONDING <ls_sel> TO lr_location_1_line.
  APPEND lr_location_1_line TO lr_location_1.
WHEN 'LOCATION2'.
  MOVE-CORRESPONDING <ls_sel> TO lr_location_2_line.
  APPEND lr_location_2_line TO lr_location_2.
WHEN 'LOCATION3'.
  MOVE-CORRESPONDING <ls_sel> TO lr_location_3_line.
  APPEND lr_location_3_line TO lr_location_3.
WHEN 'MARST'.
  MOVE-CORRESPONDING <ls_sel> TO lr_marst_line.
  APPEND lr_marst_line      TO lr_marst.
WHEN 'MCNAME1'.
  MOVE-CORRESPONDING <ls_sel> TO lr_mc_name1_line.
  APPEND lr_mc_name1_line TO lr_mc_name1.
WHEN 'MCNAME2'.
  MOVE-CORRESPONDING <ls_sel> TO lr_mc_name2_line.
  APPEND lr_mc_name2_line TO lr_mc_name2.
WHEN 'REF_CENTR'.
  MOVE-CORRESPONDING <ls_sel> TO lr_namcountry_line.
  APPEND lr_namcountry_line TO lr_namcountry.
WHEN 'REFCENTRL'.
  MOVE-CORRESPONDING <ls_sel> TO lr_nameformat_line.
  APPEND lr_nameformat_line TO lr_nameformat.
WHEN 'NAM_MIDDLE'.
  MOVE-CORRESPONDING <ls_sel> TO lr_namemiddle_line.
  APPEND lr_namemiddle_line TO lr_namemiddle.
WHEN 'NAM_FIRST'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_first_line.
  APPEND lr_name_first_line TO lr_name_first.
WHEN 'NAME_GRP1'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_grp1_line.
  APPEND lr_name_grp1_line TO lr_name_grp1.
WHEN 'NAME_GRP2'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_grp2_line.
  APPEND lr_name_grp2_line TO lr_name_grp2.
WHEN 'NAME_LAST'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_last_line.
  APPEND lr_name_last_line TO lr_name_last.
WHEN 'NAM_LAST2'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_last2_line.
  APPEND lr_name_last2_line TO lr_name_last2.

```

```

WHEN 'NAME_LST2'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_lst2_line.
  APPEND lr_name_lst2_line      TO lr_name_lst2.
WHEN 'NAME_ORG1'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_org1_line.
  APPEND lr_name_org1_line     TO lr_name_org1.
WHEN 'NAME_ORG2'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_org2_line.
  APPEND lr_name_org2_line     TO lr_name_org2.
WHEN 'NAME_ORG3'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_org3_line.
  APPEND lr_name_org3_line     TO lr_name_org3.
WHEN 'NAME_ORG4'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name_org4_line.
  APPEND lr_name_org4_line     TO lr_name_org4.
WHEN 'NAME1_TEXT'.
  MOVE-CORRESPONDING <ls_sel> TO lr_name1_text_line.
  APPEND lr_name1_text_line    TO lr_name1_text.
WHEN 'NATIO'.
  MOVE-CORRESPONDING <ls_sel> TO lr_natio_line.
  APPEND lr_natio_line        TO lr_natio.
WHEN 'NATPERS'.
  MOVE-CORRESPONDING <ls_sel> TO lr_natpers_line.
  APPEND lr_natpers_line      TO lr_natpers.
WHEN 'NICK_NAME'.
  MOVE-CORRESPONDING <ls_sel> TO lrNickname_line.
  APPEND lrNickname_line       TO lrNickname.
WHEN 'NTRLEASED'.
  MOVE-CORRESPONDING <ls_sel> TO lr_not_released_line.
  APPEND lr_not_released_line TO lr_not_released.
WHEN 'PRTGRPTYP'.
  MOVE-CORRESPONDING <ls_sel> TO lr_partgrptyp_line.
  APPEND lr_partgrptyp_line   TO lr_partgrptyp.
WHEN 'BP_HEADER'.
  MOVE-CORRESPONDING <ls_sel> TO lr_partner_line.
  APPEND lr_partner_line       TO lr_partner.
WHEN 'PREFIX1_B'.
  MOVE-CORRESPONDING <ls_sel> TO lr_prefix1_line.
  APPEND lr_prefix1_line       TO lr_prefix1.
WHEN 'PREFIX2_B'.
  MOVE-CORRESPONDING <ls_sel> TO lr_prefix2_line.
  APPEND lr_prefix2_line       TO lr_prefix2.
WHEN 'PRINT_MOD'.
  MOVE-CORRESPONDING <ls_sel> TO lr_print_mode_line.
  APPEND lr_print_mode_line   TO lr_print_mode.
WHEN 'SOURCE'.
  MOVE-CORRESPONDING <ls_sel> TO lr_source_line.
  APPEND lr_source_line        TO lr_source.
WHEN 'TITLE_BP'.
  MOVE-CORRESPONDING <ls_sel> TO lr_title_line.
  APPEND lr_title_line         TO lr_title.
WHEN 'TITLE_ACA1'.
  MOVE-CORRESPONDING <ls_sel> TO lr_title_aca1_line.
  APPEND lr_title_aca1_line    TO lr_title_aca1.
WHEN 'TITLE_ACA2'.
  MOVE-CORRESPONDING <ls_sel> TO lr_title_aca2_line.
  APPEND lr_title_aca2_line    TO lr_title_aca2.

```

```

WHEN 'TITLE_LET'.
  MOVE-CORRESPONDING <ls_sel> TO lr_title_let_line.
  APPEND lr_title_let_line      TO lr_title_let.
WHEN 'TITLE_ROYL'.
  MOVE-CORRESPONDING <ls_sel> TO lr_title_royl_line.
  APPEND lr_title_royl_line    TO lr_title_royl.
WHEN 'XBLCK'.
  MOVE-CORRESPONDING <ls_sel> TO lr_xblk_line.
  APPEND lr_xblk_line        TO lr_xblk.
WHEN 'XDELETE'.
  MOVE-CORRESPONDING <ls_sel> TO lr_xdel_line.
  APPEND lr_xdel_line        TO lr_xdel.
WHEN 'GENDER'.
  CASE <ls_sel>-option.
    WHEN 'EQ' OR 'LE' OR 'NE' OR 'GT' OR 'LT'.
      CASE <ls_sel>-low.
        WHEN '1'.
          MOVE-CORRESPONDING <ls_sel> TO lr_xsexf_line.
          lr_xsexf_line-low           = abap_true.
          APPEND lr_xsexf_line       TO lr_xsexf.
        WHEN '2'.
          MOVE-CORRESPONDING <ls_sel> TO lr_xsexm_line.
          lr_xsexm_line-low           = abap_true.
          APPEND lr_xsexm_line       TO lr_xsexm.
        WHEN OTHERS.
      ENDCASE.
    WHEN 'BT' OR 'CP' OR 'NB' OR 'NP'.
      ef_not_supported = abap_true.
    WHEN OTHERS.
      ef_not_supported = abap_true.
  ENDCASE.
WHEN 'ZZLOB'. "CUSTOM ATTRIBUTE
  MOVE-CORRESPONDING <ls_sel> TO lr_zzlob_line.
  APPEND lr_zzlob_line TO lr_zzlob.
WHEN 'ZZSTAGE'. "CUSTOM ATTRIBUTE
  MOVE-CORRESPONDING <ls_sel> TO lr_zz1_zext_wkf1_bus_line.
  APPEND lr_zz1_zext_wkf1_bus_line TO lr_zz1_zext_wkf1_bus.
WHEN OTHERS.
  "there a selection criterion that is not supported
  ef_not_supported = abap_true.
  me->mv_prev_query_not_supported = ef_not_supported.
  RETURN.
ENDCASE.
ENDLOOP.

"select the data
CLEAR lt_but000.
SELECT * FROM but000 INTO CORRESPONDING FIELDS OF TABLE lt_but000
  WHERE augrp            IN lr_augrp
    AND birthdt           IN lr_birthdt
    AND birthdt_status    IN lr_birthdt_status
    AND birthpl            IN lr_birthpl
    AND bpext              IN lr_bpext
    AND bpkind             IN lr_bpkind
    AND bu_langu            IN lr_bu_langu
    AND bu_sort1            IN lr_bu_sort1
    AND bu_sort2            IN lr_bu_sort2

```

```

AND cndsc           IN lr_cndsc
AND contact         IN lr_contact
AND deathdht        IN lr_deathdht
AND emplo           IN lr_emplo
AND found_dat       IN lr_found_dat
AND initials         IN lr_initials
AND jobgr           IN lr_jobgr
AND langu_corr      IN lr_langu_corr
AND legal_enty      IN lr_legal_enty
AND legal_org        IN lr_legal_org
AND liquid_dat       IN lr_liquid_dat
AND location_1       IN lr_location_1
AND location_2       IN lr_location_2
AND location_3       IN lr_location_2
AND marst            IN lr_marst
AND mc_name1          IN lr_mc_name1
AND mc_name2          IN lr_mc_name2
AND namcountry        IN lr_namcountry
AND nameformat        IN lr_nameformat
AND namemiddle        IN lr_namemiddle
AND name_first         IN lr_name_first
AND name_grpl          IN lr_name_grpl
AND name_grp2          IN lr_name_grp2
AND name_last           IN lr_name_last
AND name_last2          IN lr_name_last2
AND name_lst2          IN lr_name_lst2
AND name_org1          IN lr_name_org1
AND name_org2          IN lr_name_org2
AND name_org3          IN lr_name_org3
AND name_org4          IN lr_name_org4
AND name1_text          IN lr_name1_text
AND natio              IN lr_natio
AND natpers             IN lr_natpers
AND nickname            IN lrNickname
AND not_released        IN lr_not_released
AND partgrptyp         IN lr_partgrptyp
AND partner             IN lr_partner
AND prefix1              IN lr_prefix1
AND prefix2              IN lr_prefix2
AND print_mode           IN lr_print_mode
AND source                IN lr_source
AND title                 IN lr_title
AND title_aca1           IN lr_title_aca1
AND title_aca2           IN lr_title_aca2
AND title_let              IN lr_title_let
AND title_royl             IN lr_title_royl
AND xblk                 IN lr_xblk
AND xdelete              IN lr_xdelete
AND xsexf                  IN lr_xsexf
AND xsexm                  IN lr_xsexm
AND zzlob                  IN lr_zzlob "CUSTOM ATTRIBUTE"
AND zz1_zext_wkf1_bus     IN lr_zz1_zext_wkf1_bus. "CUSTOM ATTRIBUTE
IF lt_but000 IS INITIAL.
  RETURN.
ENDIF.

"map selected data to the result table

```

```

CREATE DATA lr_data LIKE LINE OF et_data.
IF lr_data IS BOUND.
  ASSIGN lr_data->* TO <ls_data>.
ENDIF.
IF <ls_data> IS NOT ASSIGNED.
  RETURN.
ENDIF.
CLEAR lt_value.
LOOP AT lt_but000 ASSIGNING <ls_but000>.
  "type-casted mapping
  gs_mapper-direction = gc_mapper_direction-query.
  gs_mapper-entity = if_mdg_bp_constants=>gc_bp_model_entity-central.
  gs_mapper-table = if_mdg_bp_constants=>gc_bp_table-bp.
  go_mapper->map(
    EXPORTING is_source_structure = <ls_but000>
    CHANGING cs_target_structure = <ls_data> ).
  CLEAR gs_mapper.
  "special handling for the gender
  ASSIGN COMPONENT 'GENDER' OF STRUCTURE <ls_data> TO <lv_value>.
  IF sy-subrc = 0.
    IF <ls_but000>-xsexf = abap_true AND <ls_but000>-xsexm = abap_false.
      <lv_value> = '1'.
    ELSEIF <ls_but000>-xsexf = abap_false AND <ls_but000>-xsexm = abap_true.
      <lv_value> = '2'.
    ELSE.
      CLEAR <lv_value>.
    ENDIF.
  ENDIF.
  "collect mapped data
  INSERT <ls_data> INTO TABLE et_data.
  lv_value = <ls_but000>-partner.
  INSERT lv_value INTO TABLE lt_value.
ENDLOOP.

"filter the result
me->filter_query_result(
  EXPORTING it_value = lt_value
  CHANGING ct_data = et_data ).

WHEN OTHERS.
  "means the query is not handled by this class, but maybe other handlers
  CLEAR ef_handled.
ENDCASE.

IF lt_value IS NOT INITIAL AND et_data IS NOT INITIAL.
  me->filter_query_result(
    EXPORTING it_value = lt_value
    CHANGING ct_data = et_data ).
ENDIF.

IF ef_handled = abap_true.
  me->mv_prev_query_not_supported = ef_not_supported.
ENDIF.
ENDMETHOD.

```

