



PUBLIC

How-To: Extend MDG-S/MDG-C by a New Entity Type (Flex Option)

Applicable Releases:

From EHP6 for SAP ERP 6.0 and from S/4HANA 1511

Version 2.0

July 2023



Document History

Document Version	Description
1.0	First official release of this guide (March 2012)
1.1	Update (August 2020)
2.0	Layout update (July 2023)



1 BUSINESS SCENARIO.....	4
1.1 Overview.....	4
1.2 Technical Background	5
1.3 Implementation	5
2 STEP BY STEP EXPLANATION	6
2.1 Extending the Data Model.....	6
2.1.1 Create a new Entity Type	6
2.1.2 Generate MDG Data Model-Specific Structures	12
2.2 Extending the UI Configuration.....	15
2.2.1 Extend the genIL (Generic Interaction Layer) Model	15
2.2.2 Connect the MDG Data Model with the genIL Data Model	20
2.2.3 Copy standard MDG-UI configurations	22
2.2.4 Create List UIBB	26
2.2.5 Replace OVP Component in Application Configuration for Supplier	32
2.2.6 Add Custom List UIBB to Supplier OVP	33
2.3 Testing the Configuration.....	37
3 ADDITIONAL INFORMATION	39
3.1 Further Reading	39
3.1.1 Information on SAP MDG on SAP S/4HANA	39
3.1.2 SAP Roadmap Explorer	39
3.1.3 Related Information	39
3.2 SAP Notes.....	39

1 Business Scenario

SAP Master Data Governance (MDG) 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.

The processes are workflow-driven and can include several approval and revision phases, and the collaboration of all users participating in the master data maintenance.

MDG offers change request (CR)-based processing of master data with integrated workflow, staging, approval, activation, and distribution.

This How-To Guide describes how to extend the MDG-S /MDG-C data model by a new entity type. The attributes of the new entity type only exist in the MDG context and not in the ERP data models (flex option).

Note:

This guide describes an extension of the standard MDG Data Model, where data will be stored in **MDG tables** after activation. This is not the right guide for you if you need an extension where the data is stored in tables outside of MDG (for example Partner Functions).

1.1 Overview

You want to extend the MDG data model for Business Partner by an additional entity type: **Purchasing Info Record**. You want this entity type to have a 1: N relationship with the Business Partner.

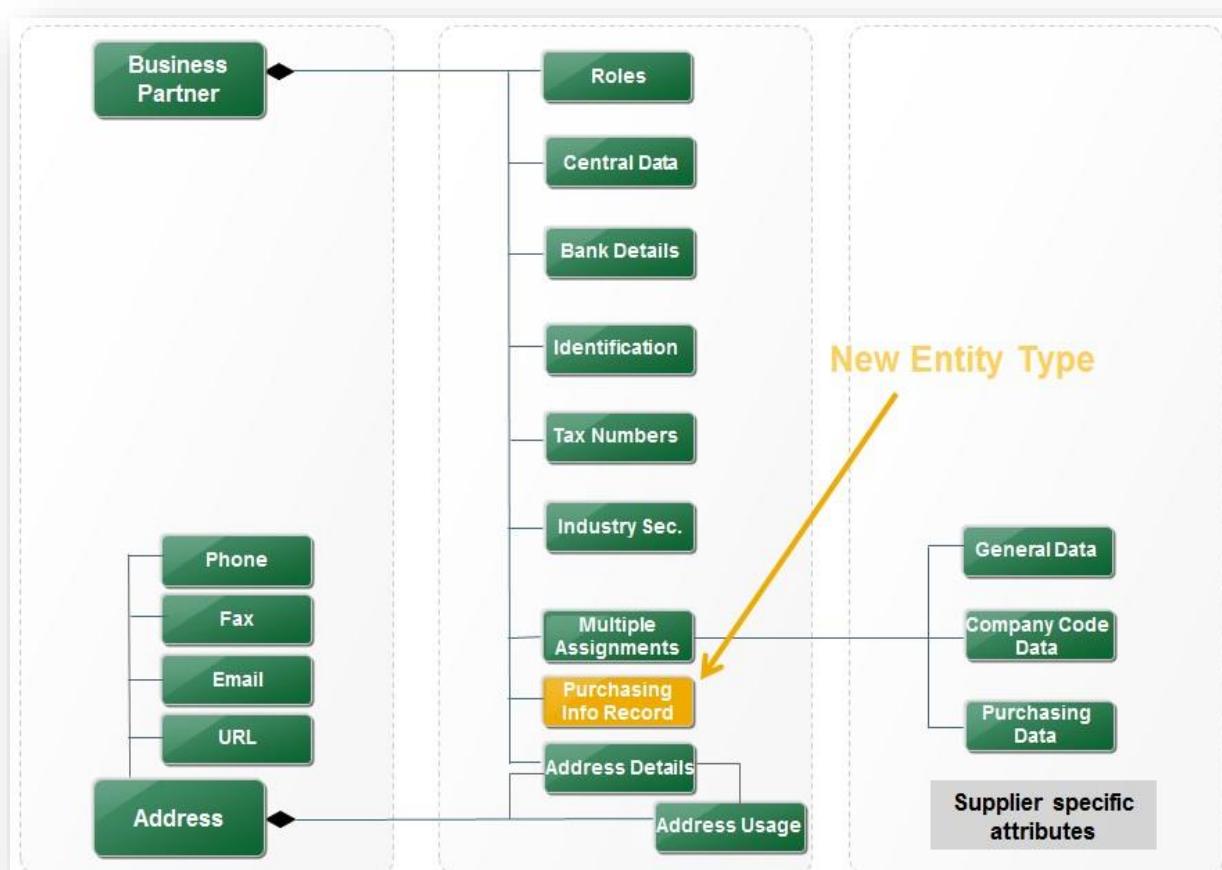


Figure: Data Model – Supplier (Scope of 2011 Delivery) with custom entity type “Purchasing Info Record”

1.2 Technical Background

This is a simple data model extension because the new field is only stored in MDG. There is no need to commit any data to ERP tables after activation of the change request. The activated data always remains in MDG and can only be used in this context.

1.3 Implementation

Two major building blocks make up the implementation of the entity type extension. In the first phase, you extend the MDG data model. In the second phase, you extend the User Interface to include the new entity type.

The flow diagram below shows the detailed implementation steps. Each box in the graphic corresponds to a section of the guide that includes detailed execution instructions.

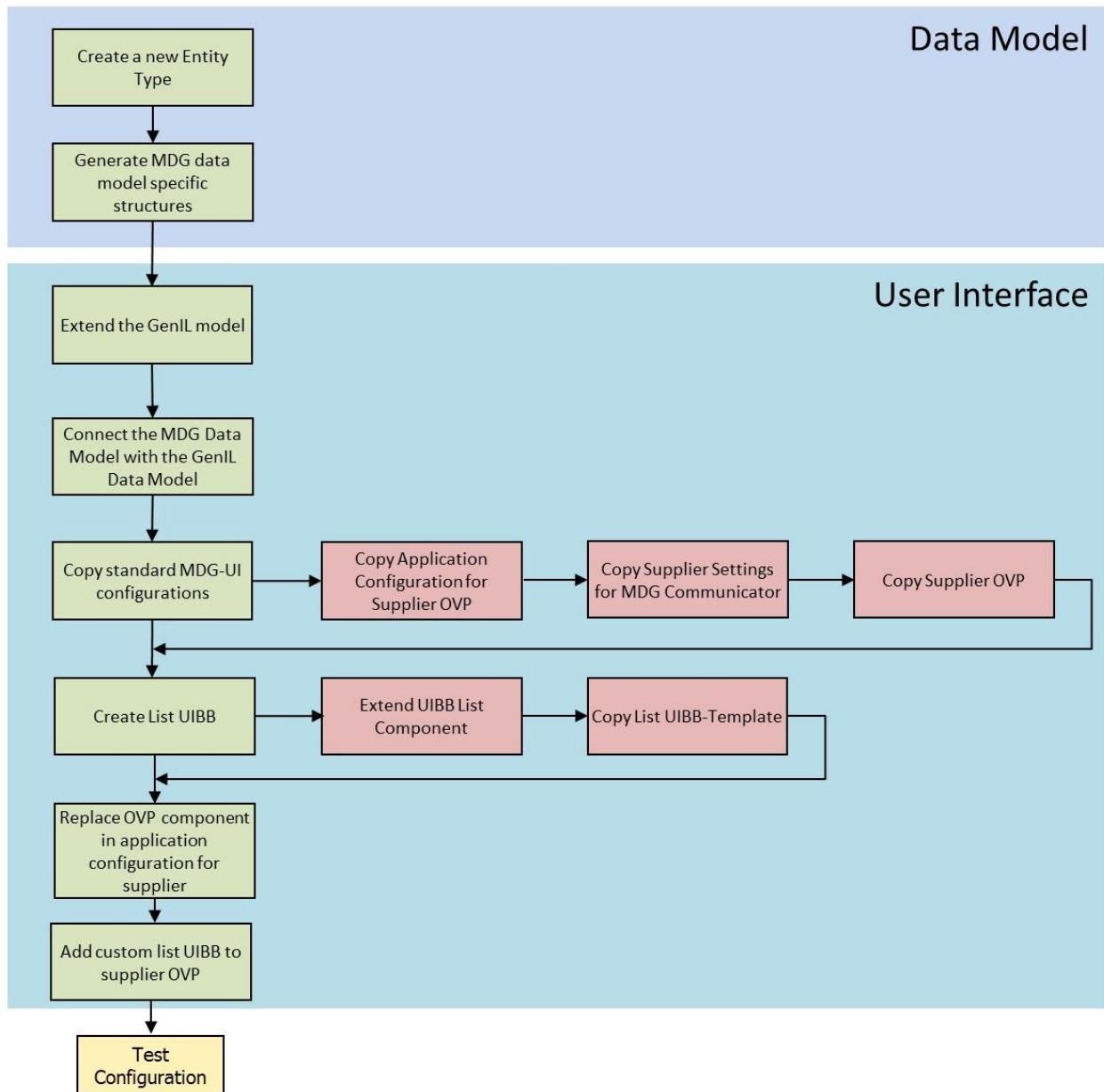


Figure: Implementation steps for flex extension

2 Step by Step Explanation

2.1 Extending the Data Model

You want to extend the MDG data model for Business Partner by an additional entity type **Purchasing Info Record**. You want the entity type to have a 1: N relationship with the Business Partner. The figure below shows how the data model looks in MDG.

You will first create a new Entity Type **ZINFOREC**. The Entity Type has two attributes **ZZPURBLOC** and **MATNR**. The relationship between **BP_HEADER** and **ZINFOREC** is 1: N of type Leading. There is a qualifying 1: N relationship between **ZINFOREC** and **MATNR**.

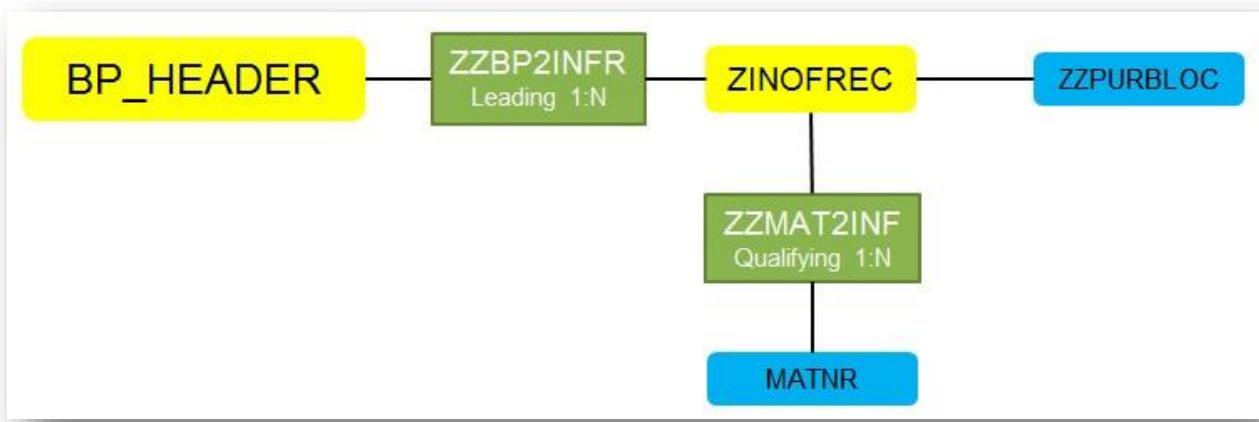


Figure: Data Model details for extension

2.1.1 Create a new Entity Type

1. Log into system for cross client maintenance.	
2. Start Customizing for Master Data Governance (Transaction MDGIMG). Go to General Settings -> Data Modeling -> Edit Data Model . Select data model BP . Double click on entity types. Click button New Entries .	<p>The screenshot shows the SAP MDGIMG interface with the title bar "Change View 'Entity Types': Overview". The toolbar includes icons for Table View, Process, Goto, Selection, Utilities, System, and Help. Below the toolbar is a menu bar with "File", "Table View", "Process", "Goto", "Selection", "Utilities", "System", and "Help". The main area is titled "Entity Types" and contains a table with columns "Entity Type", "Storage/Use Type", and "Value". The table lists entities like ADDRESS, ADDRNO, AD_CNTRY, etc. On the left, there is a "Dialog Structure" tree view with nodes like "Inactive Data Models", "Entity Types", "Attributes", "Business Object", "Entity Types for", "Hierarchy At", and "Relationships". A red arrow points to the "New Entries" button in the toolbar.</p>

3. Create New Entity Type

Entity Type: ZINFOREC

Storage/Use Type:

Changeable via Other Entity Type

Reuse Area: MDG

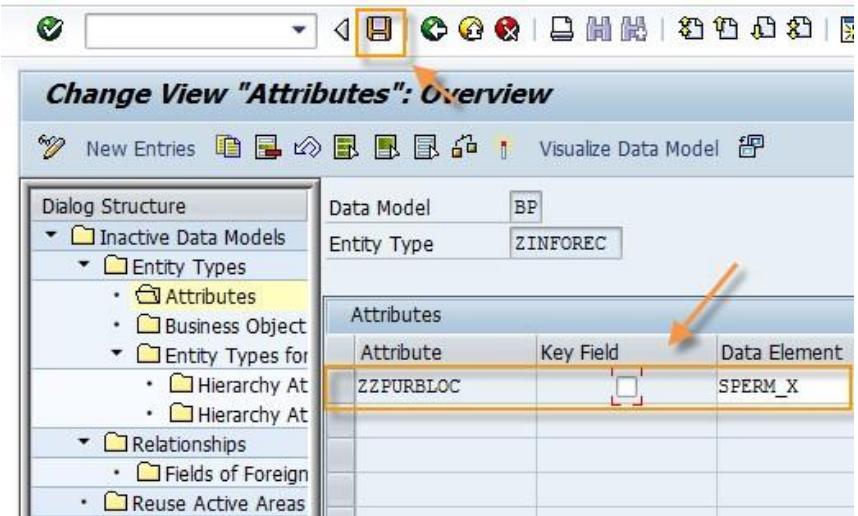
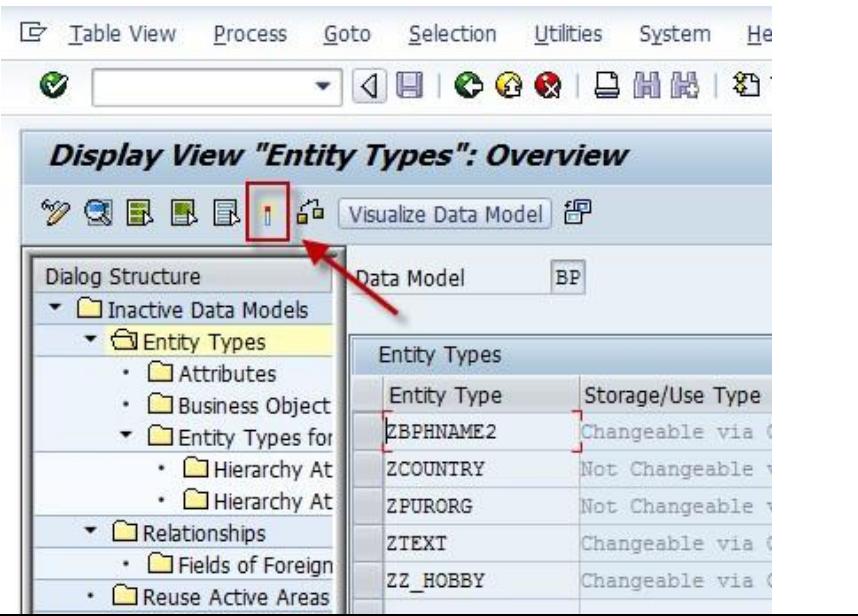
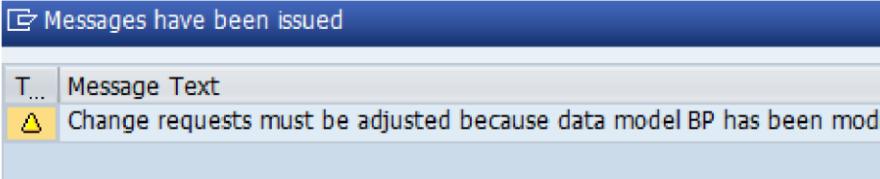
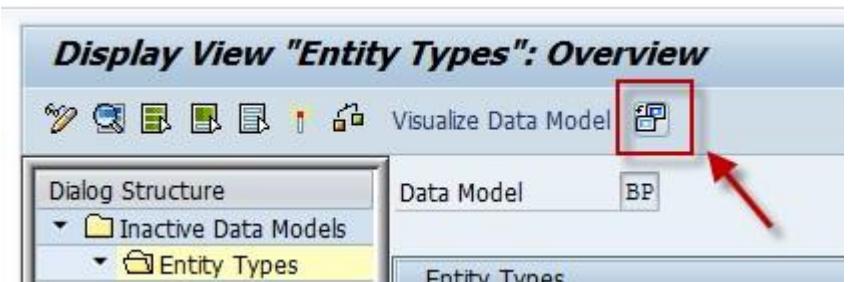
Save your settings.

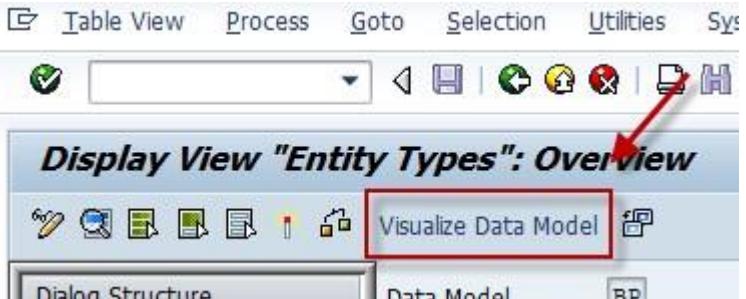
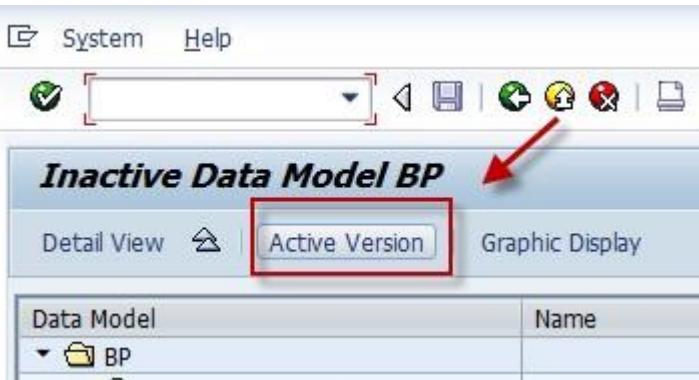
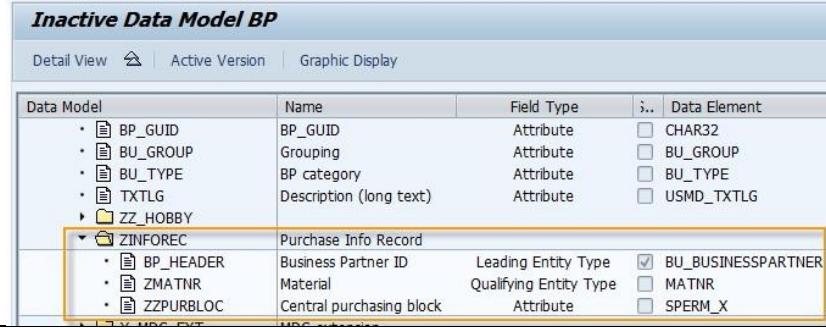
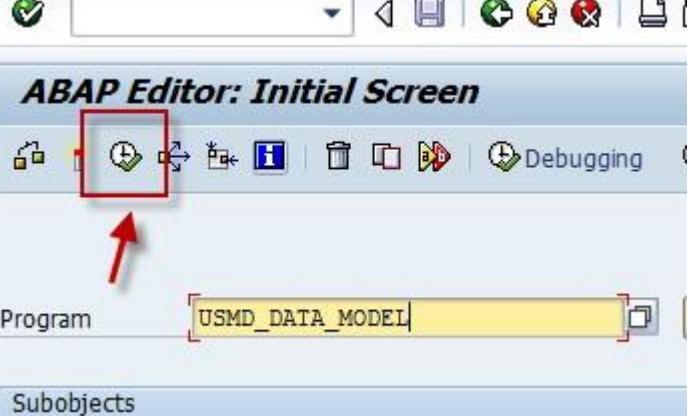
Screenshot of the SAP GUI "Change View 'Entity Types': Details" dialog box. The dialog shows the configuration for Entity Type ZINFOREC. The "Storage/Use Type" field is set to "Changeable via Other Entity Type; Generated Database Tables". The "Reuse Area" field is set to "MDG". Arrows point from the text descriptions on the left to the corresponding fields in the dialog box.

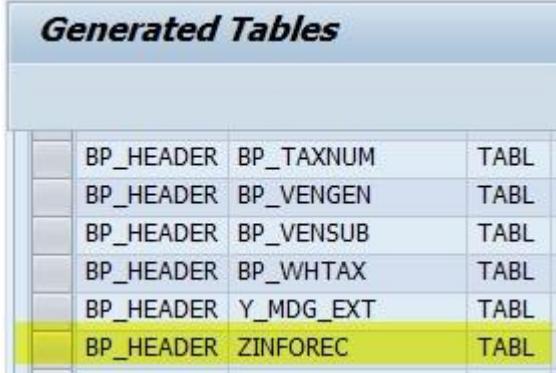
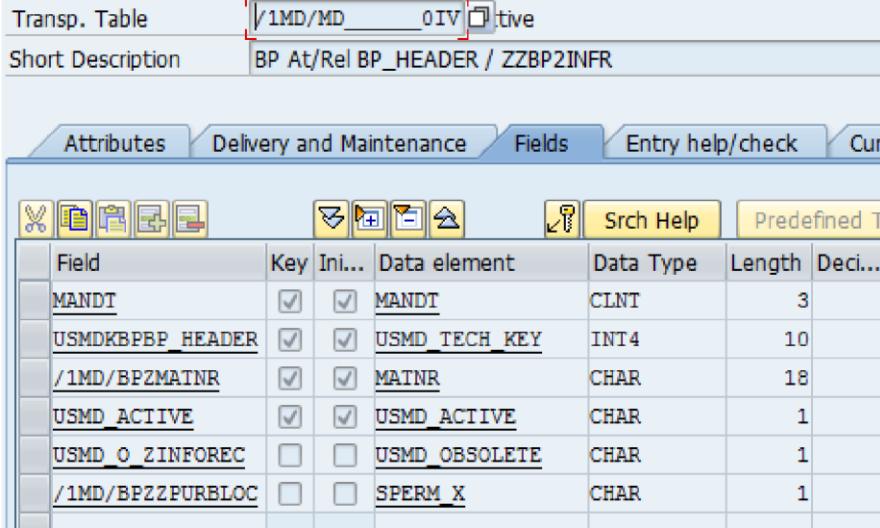
Setting	Value
Entity Type	ZINFOREC
Storage/Use Type	Changeable via Other Entity Type; Generated Database Tables
Reuse Area	MDG

<p>4. Create another New Entity Type</p> <p>Entity Type: ZMATNR</p> <p>Storage/Use Type: Not Changeable via MDG</p> <p>Data Element: MATNR</p> <p>Save your settings.</p>	
<p>5. Navigate to the Relationships node.</p> <p>Click pushbutton New Entries to create a new relationship.</p>	
<p>From-Entity Type: BP_HEADER</p> <p>Relationship: ZZBP2INFR</p> <p>To-Entity Type: ZINFOREC</p> <p>Relat. Type: Leading</p> <p>Cardinality: 1 : N</p> <p>Save your changes.</p>	

<p>7. Click pushbutton New Entries to create a new relationship.</p> <p>From-Entity Type: ZMATNR</p> <p>Relationship: ZZMAT2INF</p> <p>To-Entity Type: ZINFOREC</p> <p>Relat. Type: Qualifying</p> <p>Cardinality: 1 : N</p> <p>Save your changes.</p>	<table border="1"> <thead> <tr> <th>From-EntityType</th> <th>Relationship</th> <th>To-EntityType</th> <th>Relation. Type</th> <th>Cardinal</th> </tr> </thead> <tbody> <tr> <td>ZCOUNTRY</td> <td>ZCOUN2HN</td> <td>ZBPHNAME</td> <td>Leading</td> <td>1 : N</td> </tr> <tr> <td>ZCOUNTRY</td> <td>ZCOUN2ZHI</td> <td>ZBPHIER</td> <td>Leading</td> <td>1 : N</td> </tr> <tr> <td>ZMATNR</td> <td>ZZMAT2INF</td> <td>ZINFOREC</td> <td>Qualifying</td> <td>1 : N</td> </tr> <tr> <td>ZPURORG</td> <td>ZPO2HIER2</td> <td>ZBPHIER2</td> <td>Leading</td> <td>1 : N</td> </tr> <tr> <td>ZPURORG</td> <td>ZPO2HNAM2</td> <td>ZBPHNAME2</td> <td>Leading</td> <td>1 : N</td> </tr> </tbody> </table>	From-EntityType	Relationship	To-EntityType	Relation. Type	Cardinal	ZCOUNTRY	ZCOUN2HN	ZBPHNAME	Leading	1 : N	ZCOUNTRY	ZCOUN2ZHI	ZBPHIER	Leading	1 : N	ZMATNR	ZZMAT2INF	ZINFOREC	Qualifying	1 : N	ZPURORG	ZPO2HIER2	ZBPHIER2	Leading	1 : N	ZPURORG	ZPO2HNAM2	ZBPHNAME2	Leading	1 : N
From-EntityType	Relationship	To-EntityType	Relation. Type	Cardinal																											
ZCOUNTRY	ZCOUN2HN	ZBPHNAME	Leading	1 : N																											
ZCOUNTRY	ZCOUN2ZHI	ZBPHIER	Leading	1 : N																											
ZMATNR	ZZMAT2INF	ZINFOREC	Qualifying	1 : N																											
ZPURORG	ZPO2HIER2	ZBPHIER2	Leading	1 : N																											
ZPURORG	ZPO2HNAM2	ZBPHNAME2	Leading	1 : N																											
<p>8. Select the new Entity Type ZINFOREC and double-click on the Attributes view.</p>	<table border="1"> <thead> <tr> <th>Entity Type</th> <th>Storage/Use Type</th> </tr> </thead> <tbody> <tr> <td>ZBPHNAME2</td> <td>Changeable via Ch.</td> </tr> <tr> <td>ZCOUNTRY</td> <td>Not Changeable via</td> </tr> <tr> <td>ZINFOREC</td> <td>Changeable via Otl</td> </tr> <tr> <td>ZMATNR</td> <td>Not Changeable via</td> </tr> <tr> <td>ZPURORG</td> <td>Not Changeable via</td> </tr> <tr> <td>ZTEXT</td> <td>Changeable via Ch.</td> </tr> <tr> <td>ZZ_HOBBY</td> <td>Changeable via Otl</td> </tr> </tbody> </table>	Entity Type	Storage/Use Type	ZBPHNAME2	Changeable via Ch.	ZCOUNTRY	Not Changeable via	ZINFOREC	Changeable via Otl	ZMATNR	Not Changeable via	ZPURORG	Not Changeable via	ZTEXT	Changeable via Ch.	ZZ_HOBBY	Changeable via Otl														
Entity Type	Storage/Use Type																														
ZBPHNAME2	Changeable via Ch.																														
ZCOUNTRY	Not Changeable via																														
ZINFOREC	Changeable via Otl																														
ZMATNR	Not Changeable via																														
ZPURORG	Not Changeable via																														
ZTEXT	Changeable via Ch.																														
ZZ_HOBBY	Changeable via Otl																														
<p>9. Choose the Edit pushbutton.</p> <p>Choose the New Entries pushbutton.</p> <p>Create a new attribute with the following values:</p> <p>Attribute: ZZPURBLOC</p> <p>Data Element: SPERM_X</p>	<table border="1"> <thead> <tr> <th>Attribute</th> <th>Key Field</th> <th>Data Element</th> </tr> </thead> <tbody> <tr> <td>ZZPURBLOC</td> <td></td> <td>SPERM_X</td> </tr> </tbody> </table>	Attribute	Key Field	Data Element	ZZPURBLOC		SPERM_X																								
Attribute	Key Field	Data Element																													
ZZPURBLOC		SPERM_X																													

10. Save your changes.	
11. Activate your data model changes.	
12. A messages popup displays (see screenshot.)	
13. Choose the Adjust Staging Area as shown to adjust existing change requests.	

14. In the following steps, you verify if the MDG staging structures were successfully generated. Choose the Visualize Data Model pushbutton.																																									
15. Choose the Active Version pushbutton.																																									
16. The screenshot shows the generated structures.	 <table border="1"> <thead> <tr> <th>Data Model</th> <th>Name</th> <th>Field Type</th> <th>Data Element</th> </tr> </thead> <tbody> <tr> <td>BP_GUID</td> <td>BP_GUID</td> <td>Attribute</td> <td>CHAR32</td> </tr> <tr> <td>BU_GROUP</td> <td>Grouping</td> <td>Attribute</td> <td>BU_GROUP</td> </tr> <tr> <td>BU_TYPE</td> <td>BP category</td> <td>Attribute</td> <td>BU_TYPE</td> </tr> <tr> <td>TXTLG</td> <td>Description (long text)</td> <td>Attribute</td> <td>USMD_TXTLG</td> </tr> <tr> <td>ZZ_HOBBY</td> <td></td> <td></td> <td></td> </tr> <tr> <td>ZINFOREC</td> <td>Purchase Info Record</td> <td></td> <td></td> </tr> <tr> <td> BP_HEADER</td> <td>Business Partner ID</td> <td>Leading Entity Type</td> <td>BU_BUSINESSPARTNER</td> </tr> <tr> <td> ZMATNR</td> <td>Material</td> <td>Qualifying Entity Type</td> <td>MATNR</td> </tr> <tr> <td> ZZPURBLOC</td> <td>Central purchasing block</td> <td>Attribute</td> <td>SPERM_X</td> </tr> </tbody> </table>	Data Model	Name	Field Type	Data Element	BP_GUID	BP_GUID	Attribute	CHAR32	BU_GROUP	Grouping	Attribute	BU_GROUP	BU_TYPE	BP category	Attribute	BU_TYPE	TXTLG	Description (long text)	Attribute	USMD_TXTLG	ZZ_HOBBY				ZINFOREC	Purchase Info Record			BP_HEADER	Business Partner ID	Leading Entity Type	BU_BUSINESSPARTNER	ZMATNR	Material	Qualifying Entity Type	MATNR	ZZPURBLOC	Central purchasing block	Attribute	SPERM_X
Data Model	Name	Field Type	Data Element																																						
BP_GUID	BP_GUID	Attribute	CHAR32																																						
BU_GROUP	Grouping	Attribute	BU_GROUP																																						
BU_TYPE	BP category	Attribute	BU_TYPE																																						
TXTLG	Description (long text)	Attribute	USMD_TXTLG																																						
ZZ_HOBBY																																									
ZINFOREC	Purchase Info Record																																								
BP_HEADER	Business Partner ID	Leading Entity Type	BU_BUSINESSPARTNER																																						
ZMATNR	Material	Qualifying Entity Type	MATNR																																						
ZZPURBLOC	Central purchasing block	Attribute	SPERM_X																																						
17. To view the generated tables, start transaction SE38 . Enter program USMD_DATA_MODEL . Enter data model BP . Run the program.																																									

18. Double click on ZINFOREC																																																		
19. Your table should look similar to the one shown in the screenshot.	 <table border="1" data-bbox="568 781 1448 1107"> <thead> <tr> <th>Field</th> <th>Key</th> <th>Ini...</th> <th>Data element</th> <th>Data Type</th> <th>Length</th> <th>Deci...</th> </tr> </thead> <tbody> <tr> <td>MANDT</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>MANDT</td> <td>CLNT</td> <td>3</td> <td></td> </tr> <tr> <td>USMDKBPBP_HEADER</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>USMD_TECH_KEY</td> <td>INT4</td> <td>10</td> <td></td> </tr> <tr> <td>/1MD/BPZMATNR</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>MATNR</td> <td>CHAR</td> <td>18</td> <td></td> </tr> <tr> <td>USMD_ACTIVE</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>USMD_ACTIVE</td> <td>CHAR</td> <td>1</td> <td></td> </tr> <tr> <td>USMD_O_ZINFOREC</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>USMD_OBSOLETE</td> <td>CHAR</td> <td>1</td> <td></td> </tr> <tr> <td>/1MD/BPZZPURBLOC</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>SPERM_X</td> <td>CHAR</td> <td>1</td> <td></td> </tr> </tbody> </table>	Field	Key	Ini...	Data element	Data Type	Length	Deci...	MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3		USMDKBPBP_HEADER	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	USMD_TECH_KEY	INT4	10		/1MD/BPZMATNR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MATNR	CHAR	18		USMD_ACTIVE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	USMD_ACTIVE	CHAR	1		USMD_O_ZINFOREC	<input type="checkbox"/>	<input type="checkbox"/>	USMD_OBSOLETE	CHAR	1		/1MD/BPZZPURBLOC	<input type="checkbox"/>	<input type="checkbox"/>	SPERM_X	CHAR	1	
Field	Key	Ini...	Data element	Data Type	Length	Deci...																																												
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3																																													
USMDKBPBP_HEADER	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	USMD_TECH_KEY	INT4	10																																													
/1MD/BPZMATNR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MATNR	CHAR	18																																													
USMD_ACTIVE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	USMD_ACTIVE	CHAR	1																																													
USMD_O_ZINFOREC	<input type="checkbox"/>	<input type="checkbox"/>	USMD_OBSOLETE	CHAR	1																																													
/1MD/BPZZPURBLOC	<input type="checkbox"/>	<input type="checkbox"/>	SPERM_X	CHAR	1																																													

2.1.2 Generate MDG Data Model-Specific Structures

After you change the MDG data model, you must regenerate the tables. In this customizing activity, for each data model and entity type, you generate technical structures and tables in the ABAP Dictionary. The system uses these structures internally for implementing the staging area. To generate these Data Model-specific structures follow the steps below.

Note

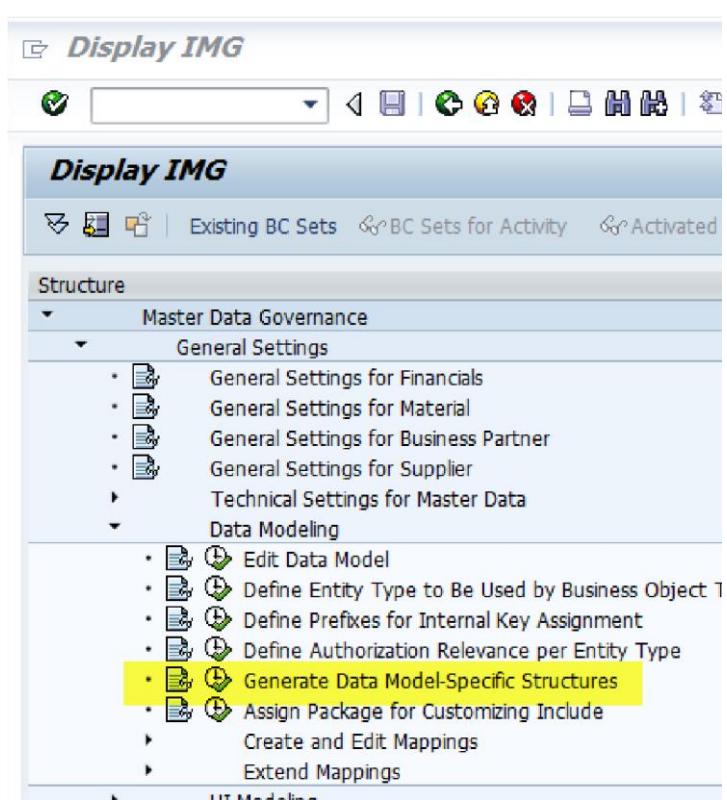
In general, if you change a data model (for example, if you change attributes of entity types or relationships), you need to regenerate the structures.

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

2. Start Customizing for **Master Data Governance**
(transaction **MDGIMG**).

Go to **General Settings -> Data Modeling**

-> **Generate Data Model-specific Structures**



3. Create two **New Entries** as shown in the screenshot.

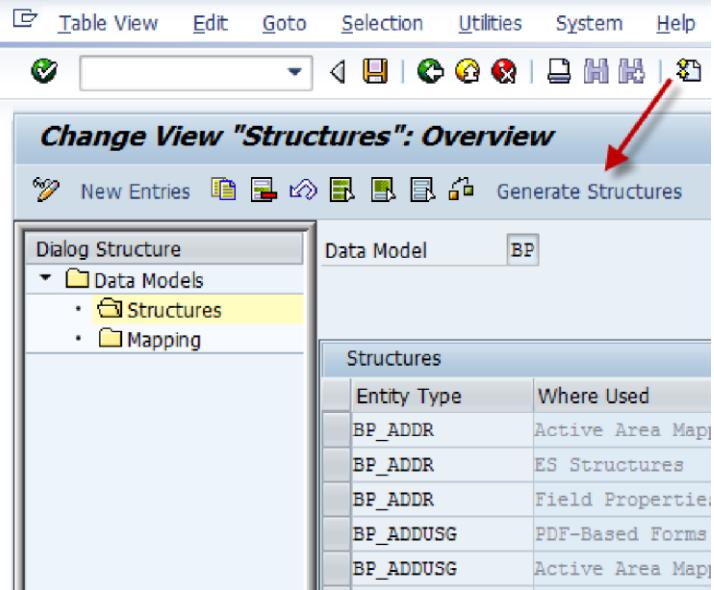
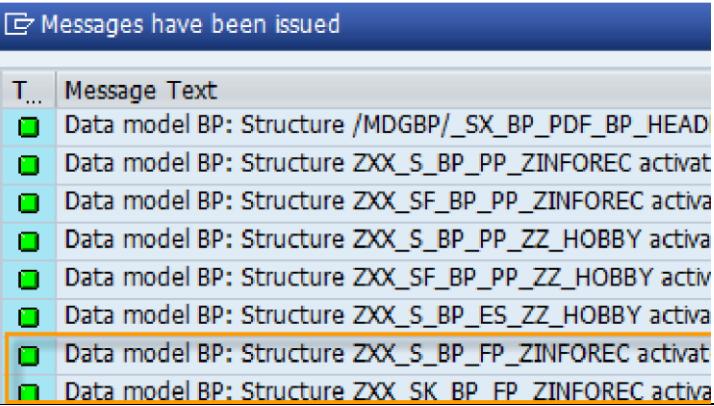
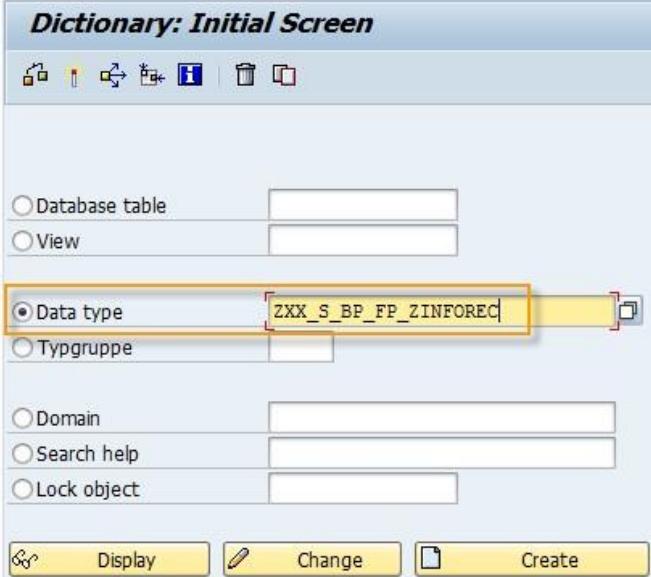
One entry for **Active Area Mapping**.

On entry for **Field Properties**.

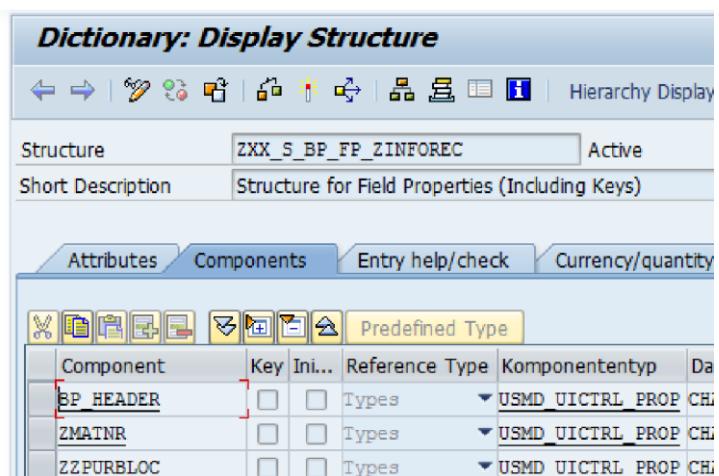
Save your changes.

The screenshot shows the 'Structures': Overview dialog. At the top, there is a toolbar with various icons. Below it, there is a table titled 'Structures' with columns: Entity Type, Where Used, Prefix/N..., Name of Struct..., and Pack...'. There are seven rows in the table. The second and third rows are highlighted with yellow boxes. The second row contains 'BP_WHTAX' in 'Entity Type', 'ES Structur...' in 'Where Used', '/MDGBP...' in 'Prefix/N...', 'ES_BP_WHTAX' in 'Name of Struct...', and '/MDGBP...' in 'Pack...'. The third row contains 'ZINFOREC' in 'Entity Type', 'Active Area...' in 'Where Used', 'ZXX' in 'Prefix/N...', 'PP_ZINFOREC' in 'Name of Struct...', and '\$TMP' in 'Pack...'. The other five rows are standard entries.

Entity Type	Where Used	Prefix/N...	Name of Struct...	Pack...
BP_WHTAX	ES Structur...	/MDGBPX/	ES_BP_WHTAX	/MDGBP...
BP_WHTAX	Field Propre...	/MDGBPX/	FP_BP_WHTAX	/MDGBP...
ZINFOREC	Active Area...	ZXX	PP_ZINFOREC	\$TMP
ZINFOREC	Field Propre...	ZXX	FP_ZINFOREC	\$TMP
ZZ_HOBBY	Active Area...	ZXX	PP_ZZ_HOBBY	\$TMP
ZZ_HOBBY	ES Structur...	ZXX	ES_ZZ_HOBBY	\$TMP
ZZ_HOBBY	Field Propre...	ZXX	FP_ZZ_HOBBY	\$TMP

4. Select the row with data model BP Double-click Structures in the left-hand panel Click Generate Structures .	
5. Verify that your structures for ZINFOREC were successfully generated.	
6. In the following steps, you verify that the active area mapping structure was successfully generated. Start transaction SE11 . Display structure ZXX_S_BP_FP_ZINFOREC by entering the details as shown.	

7.	You have now verified that the structure has been generated.
----	--



2.2 Extending the UI Configuration

To display the new fields in the Supplier-UI you first need to create a new Form-UIBB (User Interface Building Block) and add it to the UI component configuration **BS_SP_OVP**. You can use any of the following FPM options:

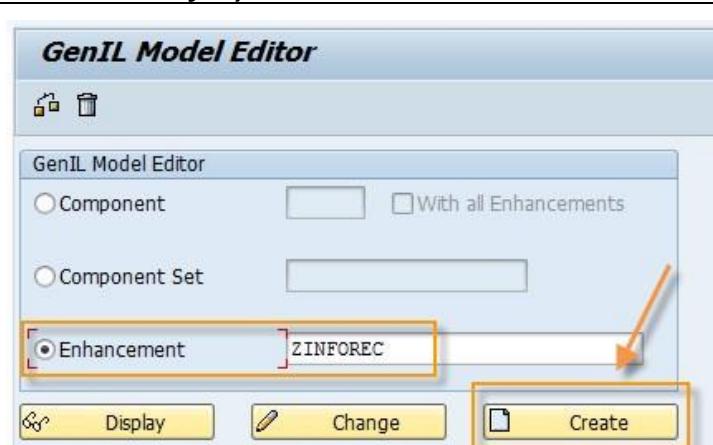
- Enhancement, adaptation, or customization of the existing component
- Adaptation of the component
- Copying of the UI configuration **ZZBS_SP_OVP** (as shown in this guide)

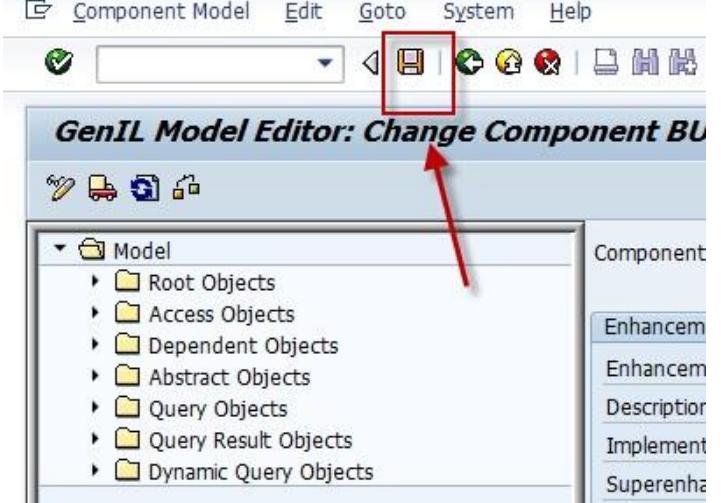
Note

For information on how to adapt the user interface using the floorplan manager; including advantages, disadvantages, and steps required, see SAP Note 1619534

2.2.1 Extend the genIL (Generic Interaction Layer) Model

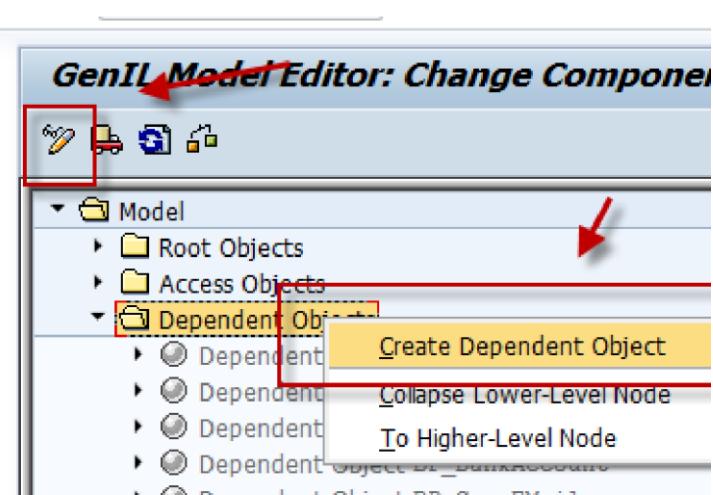
1.	Create an enhancement for the standard BUPA genIL Model (transaction code: genIL_model_browser)
----	--



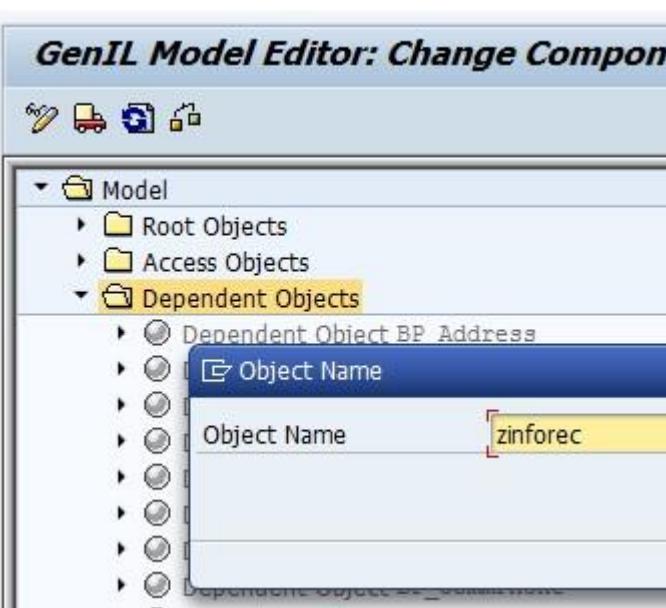
2. Enter Superenhancement : BUPA_CUSP.																									
3. Verify the details of the enhancement.	<p>Component Details</p> <table border="1"> <tr><td>Enhancement</td><td>ZINFOREC</td></tr> <tr><td>Description</td><td>Extensibility Test</td></tr> <tr><td>Implementation Class</td><td></td></tr> <tr><td>Superenhancement</td><td>BUPA_CUSP</td></tr> <tr><td>Prefix</td><td></td></tr> <tr><td><input type="checkbox"/> Final</td><td></td></tr> <tr><td>Created</td><td>16.03.2012</td></tr> <tr><td>Last Change</td><td>16.03.2012</td></tr> </table> <p>Component Settings</p> <table border="1"> <tr><td>Component</td><td>BUPA</td></tr> <tr><td>Description</td><td></td></tr> <tr><td>Implementation Class</td><td>CL_BS_GENIL_BUPA</td></tr> <tr><td>Object Table</td><td></td></tr> </table>	Enhancement	ZINFOREC	Description	Extensibility Test	Implementation Class		Superenhancement	BUPA_CUSP	Prefix		<input type="checkbox"/> Final		Created	16.03.2012	Last Change	16.03.2012	Component	BUPA	Description		Implementation Class	CL_BS_GENIL_BUPA	Object Table	
Enhancement	ZINFOREC																								
Description	Extensibility Test																								
Implementation Class																									
Superenhancement	BUPA_CUSP																								
Prefix																									
<input type="checkbox"/> Final																									
Created	16.03.2012																								
Last Change	16.03.2012																								
Component	BUPA																								
Description																									
Implementation Class	CL_BS_GENIL_BUPA																								
Object Table																									
4. Save the enhancement.																									

5. Choose the **Edit** pushbutton.

In the **Model** tree, select **Dependent Object** -> **Create Dependent Object**.



6. Enter the name of the dependent object as shown.



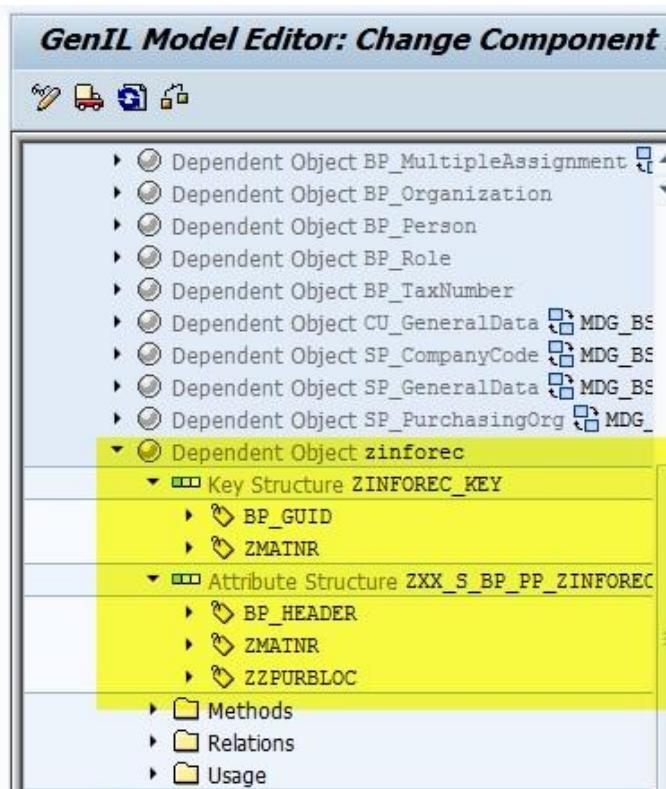
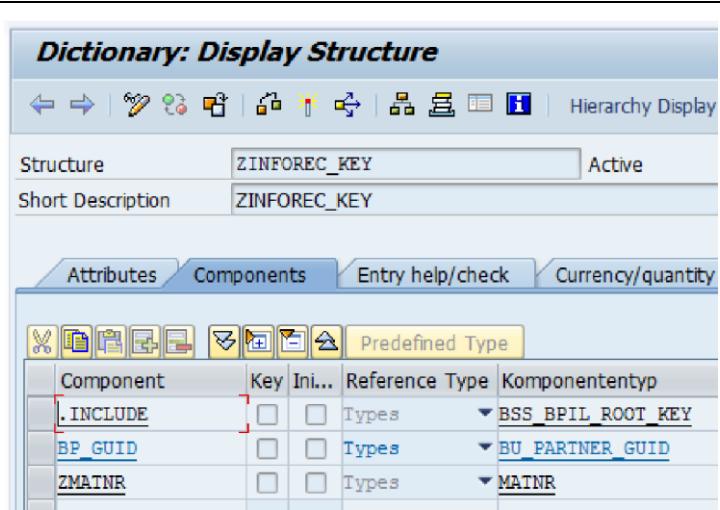
7. You must assign a **Key Structure** and an attribute structure to the dependent object.

For the **Attribute Structure**, you enter the name of the active area mapping structure that you have generated in step 23 above (**ZXX_S_BP_PP_ZINFOREC**).

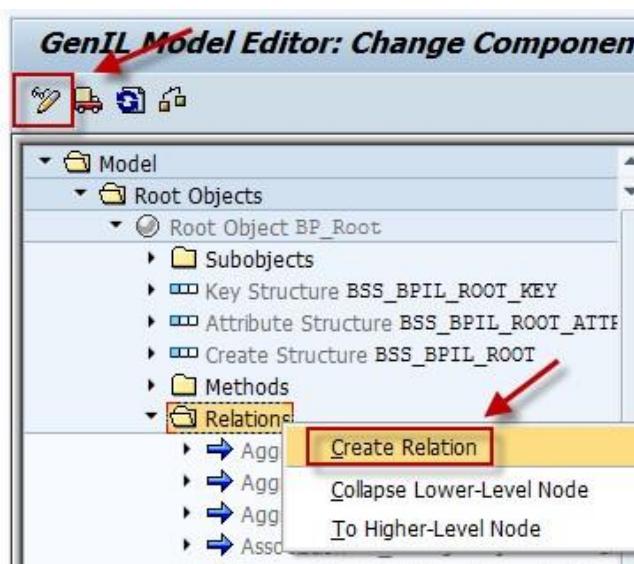
For the Key Structure you must enter the name of a structure that you create in the next step (**ZINFOREC_KEY**).

Dependent Object	
Enhancement	
Original Definition In	ZINFOREC
Enhancement Description	Extensibility Test
Displayed Definition In	ZINFOREC
Enhancement Description	Extensibility Test
Basic Settings	
Component	BUPA
Object	zinforec
Object Type	Dependent Object
Key Structure	ZINFOREC_KEY
Attribute Structure	ZXX_S_BP_PP_ZINFOREC
Root Object	BP_Root
Super Object	
Web Service Enabled	<input type="checkbox"/>

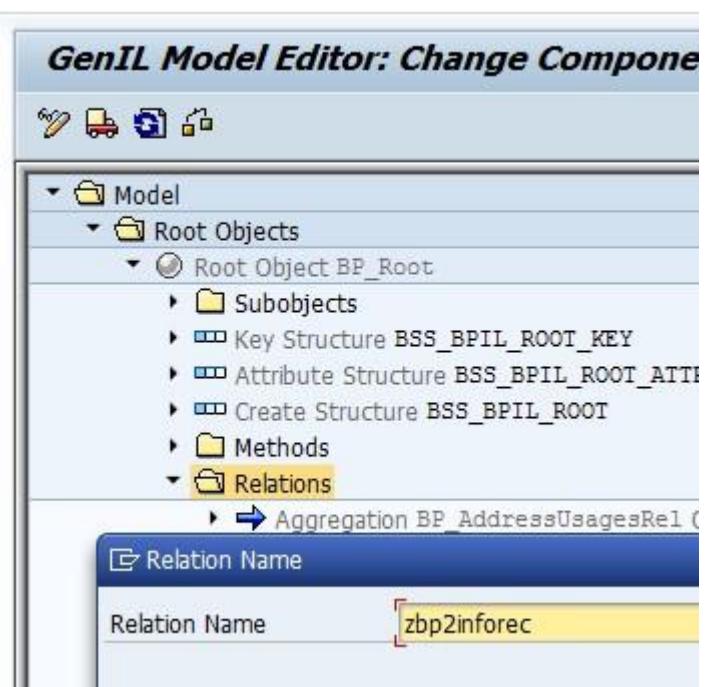
	<p>8. Start transaction SE11.</p> <p>Create structure ZINFOREC_KEY as shown.</p> <ul style="list-style-type: none"> Component: .INCLUDE Reference Type: Types Component Type: BSS_BPIL_ROOT_KEY Component: ZMATNR Reference Type: Types Component Type: MATNR <p>Save your changes. Activate the structure.</p>
9.	<p>Verify that the model nodes look similar to the ones shown.</p>

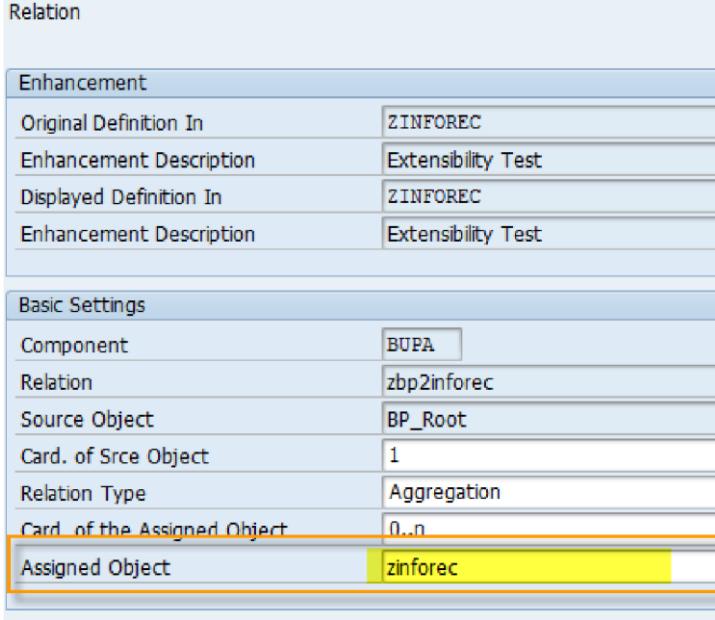
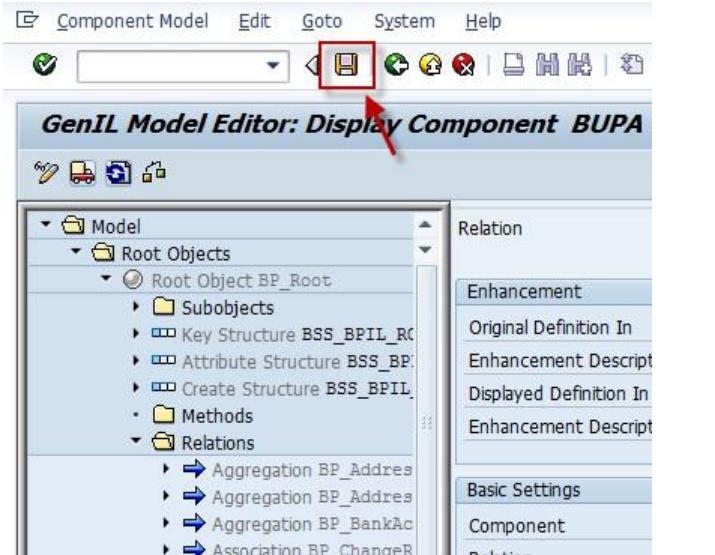


10. Switch to edit mode.
Navigate to the **Relations** node. Right-click and select **Create Relations**.



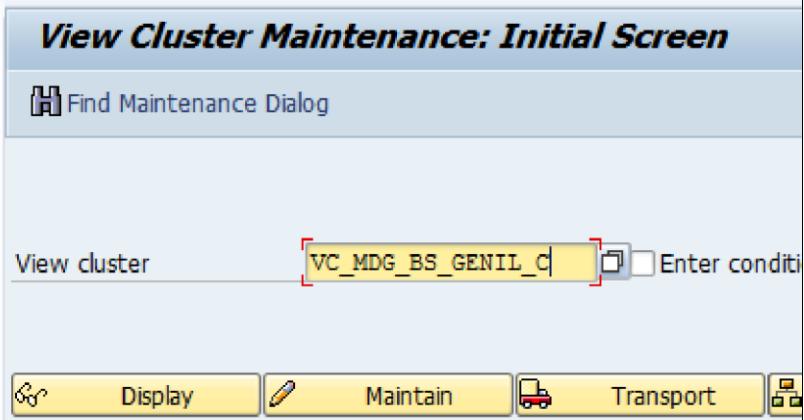
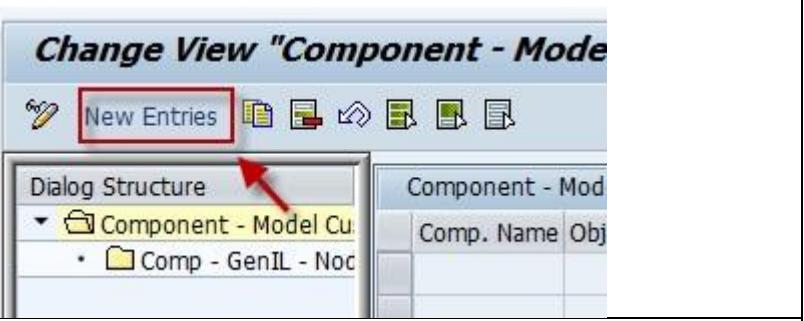
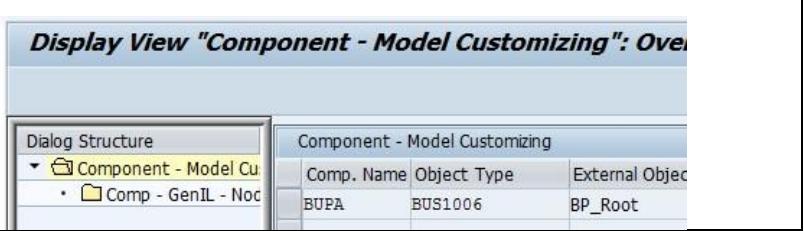
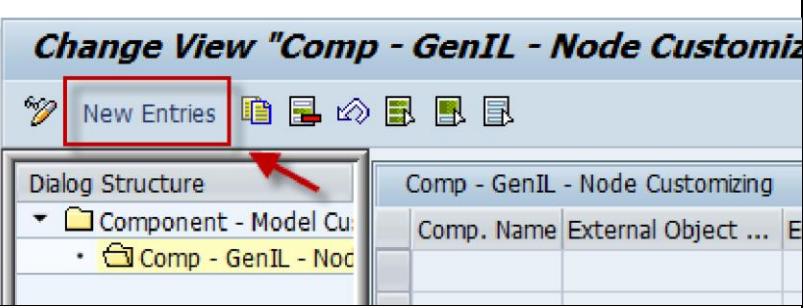
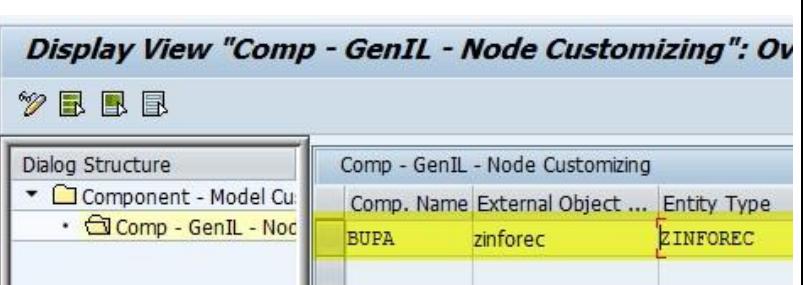
11. Enter the details as shown.



12. In the relations detail screen select the assigned object ZINFOREC .	
13. Save your changes.	

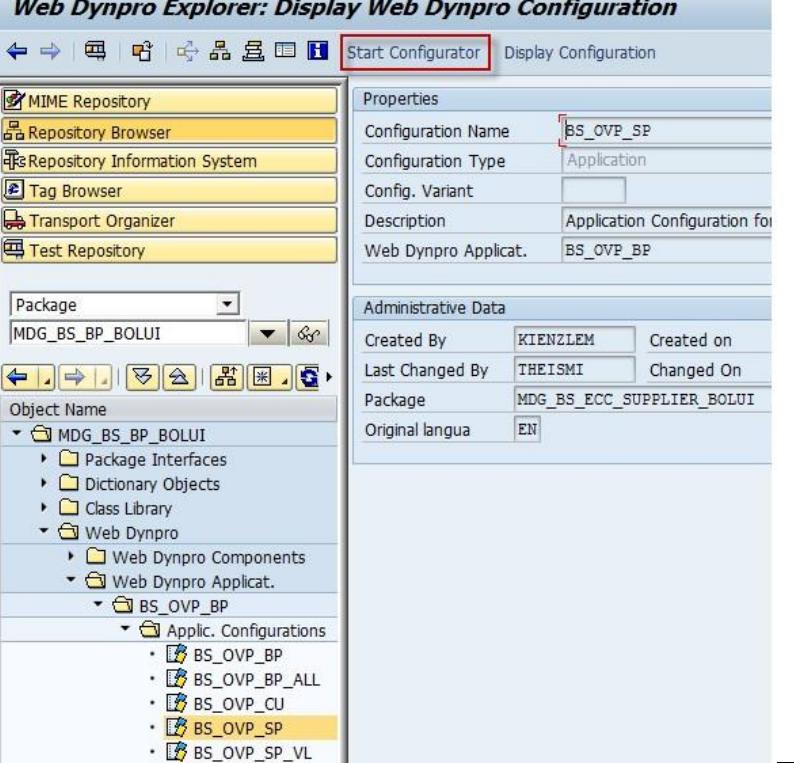
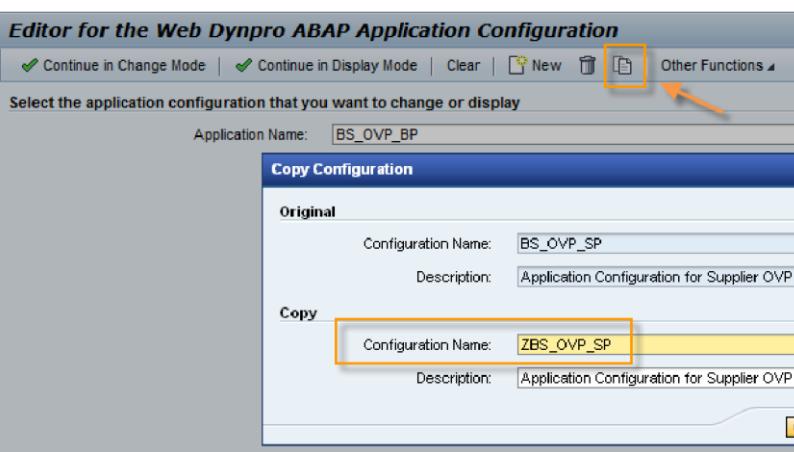
2.2.2 Connect the MDG Data Model with the genIL Data Model

Create entries in the **vc_MDG_BS_GENIL_C** view cluster to relate the genIL model component to the entity type of the MDG data model.

<p>1. Start transaction SE34. Display View Cluster VC_MDG_BS_GENIL_C</p>										
<p>2. Choose the New Entries pushbutton.</p>										
<p>3. Create a new entry with the following details: Comp. Name: BUPA Object Type: BUS1006 External Object: BP_Root</p>	 <table border="1" data-bbox="652 990 1456 1221"> <thead> <tr> <th colspan="3">Component - Model Customizing</th> </tr> <tr> <th>Comp. Name</th> <th>Object Type</th> <th>External Object</th> </tr> </thead> <tbody> <tr> <td>BUPA</td> <td>BUS1006</td> <td>BP_Root</td> </tr> </tbody> </table>	Component - Model Customizing			Comp. Name	Object Type	External Object	BUPA	BUS1006	BP_Root
Component - Model Customizing										
Comp. Name	Object Type	External Object								
BUPA	BUS1006	BP_Root								
<p>4. Mark the new entry and select Comp - GenIL - Node. Then choose the New Entries pushbutton.</p>										
<p>5. Create a new entry with the following details: Comp. Name: BUPA External Object: ZINFOREC Entity Type: ZINFOREC</p>	 <table border="1" data-bbox="652 1529 1456 1814"> <thead> <tr> <th colspan="3">Comp - GenIL - Node Customizing</th> </tr> <tr> <th>Comp. Name</th> <th>External Object ...</th> <th>Entity Type</th> </tr> </thead> <tbody> <tr> <td>BUPA</td> <td>zinforec</td> <td>ZINFOREC</td> </tr> </tbody> </table>	Comp - GenIL - Node Customizing			Comp. Name	External Object ...	Entity Type	BUPA	zinforec	ZINFOREC
Comp - GenIL - Node Customizing										
Comp. Name	External Object ...	Entity Type								
BUPA	zinforec	ZINFOREC								

2.2.3 Copy standard MDG-UI configurations

Copy Application Configuration for Supplier Overview page (OVP)

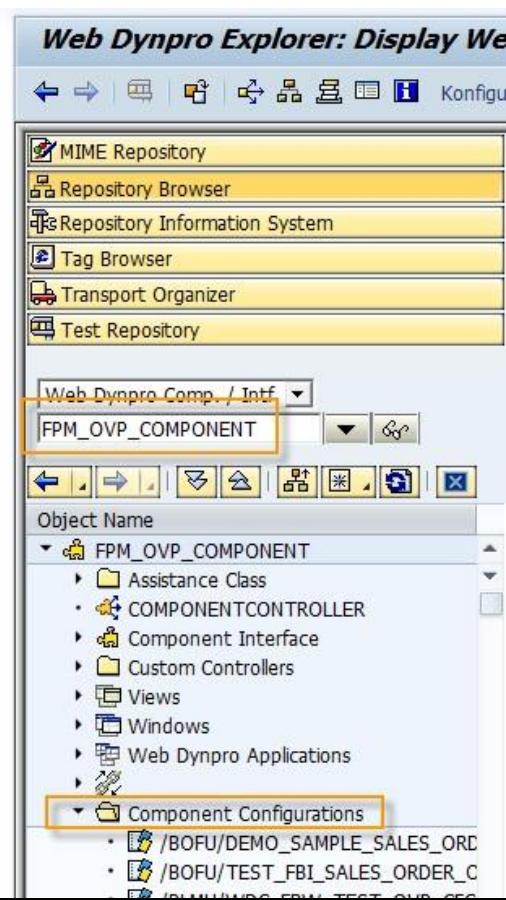
<p>1. Start transaction SE80. Navigate to the Application Configuration as shown. Choose the Start Configurator pushbutton.</p>	 <p>Web Dynpro Explorer: Display Web Dynpro Configuration</p> <p>Properties</p> <ul style="list-style-type: none"> Configuration Name: BS_OVP_SP Configuration Type: Application Config. Variant: Description: Application Configuration for Supplier Overview page (OVP) Web Dynpro Applcat.: BS_OVP_BP <p>Administrative Data</p> <ul style="list-style-type: none"> Created By: KIENZLEM Created on: Last Changed By: THEISMI Changed On: Package: MDG_BS_ECC_SUPPLIER_BOLUI Original langua: EN <p>Object Name</p> <ul style="list-style-type: none"> MDG_BS_BP_BOLUI MDG_BS_BP_BOLUI <ul style="list-style-type: none"> Package Interfaces Dictionary Objects Class Library Web Dynpro <ul style="list-style-type: none"> Web Dynpro Components Web Dynpro Applcat. <ul style="list-style-type: none"> BS_OVP_BP <ul style="list-style-type: none"> Applic. Configurations <ul style="list-style-type: none"> BS_OVP_SP (highlighted) BS_OVP_BP_ALL BS_OVP_CU BS_OVP_SP_VL 						
<p>2. Copy the existing configuration to ZBS_OVP_SP.</p>	 <p>Editor for the Web Dynpro ABAP Application Configuration</p> <p>Select the application configuration that you want to change or display</p> <p>Application Name: BS_OVP_BP</p> <p>Copy Configuration</p> <table border="1"> <thead> <tr> <th>Original</th> <th>Copy</th> </tr> </thead> <tbody> <tr> <td>Configuration Name: BS_OVP_SP</td> <td>Configuration Name: ZBS_OVP_SP (highlighted)</td> </tr> <tr> <td>Description: Application Configuration for Supplier OVP</td> <td>Description: Application Configuration for Supplier OVP</td> </tr> </tbody> </table>	Original	Copy	Configuration Name: BS_OVP_SP	Configuration Name: ZBS_OVP_SP (highlighted)	Description: Application Configuration for Supplier OVP	Description: Application Configuration for Supplier OVP
Original	Copy						
Configuration Name: BS_OVP_SP	Configuration Name: ZBS_OVP_SP (highlighted)						
Description: Application Configuration for Supplier OVP	Description: Application Configuration for Supplier OVP						

Copy Supplier Settings for MDG Communicator

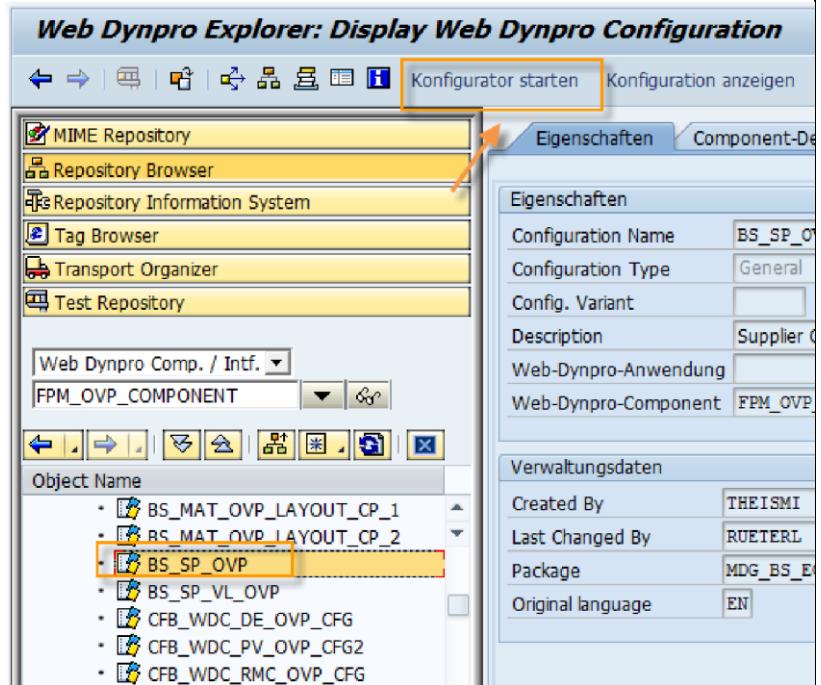
<p>1. Start transaction SE80.</p> <p>Navigate to the Component Configuration for Web Dynpro application MDG_BS_GOV_COMMUNICATOR</p> <p>Choose the Start Configurator pushbutton.</p>	
<p>2. Copy the existing configuration to ZBS_OVP_SP.</p> <p>It is important that you use the same Name as the one in the application configuration in the previous section (ZBS_OVP_SP). This makes sure that you get the change request header in your Z application.</p>	

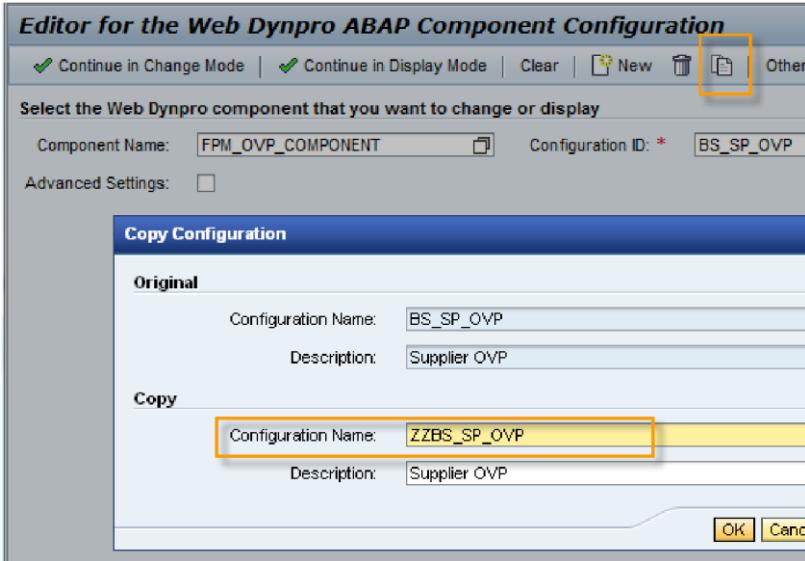
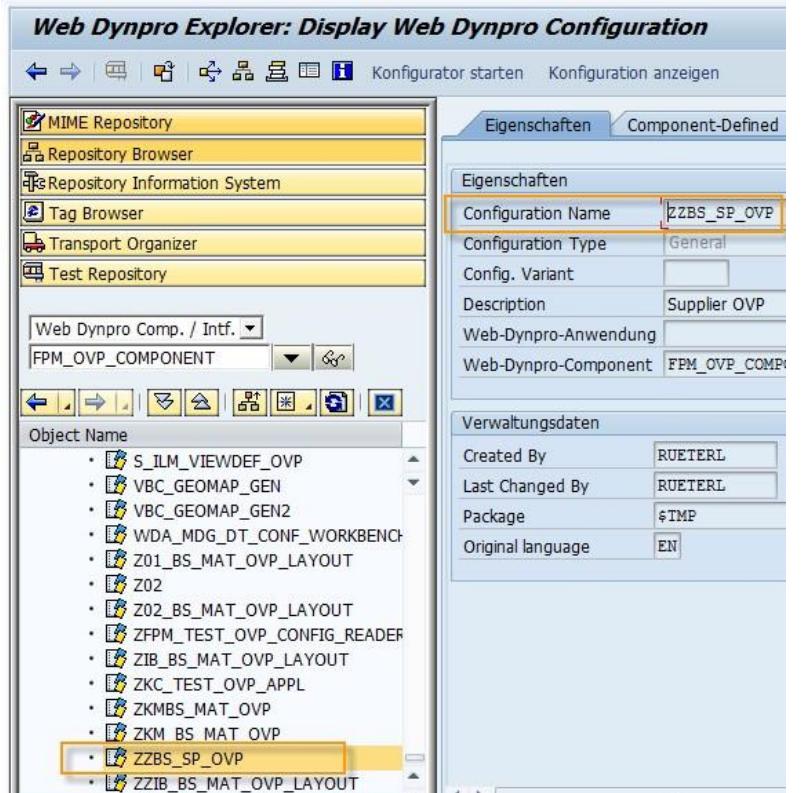
Copy Supplier OVP

- Start transaction **SE80**.
Navigate to the **Component Configuration** as shown.



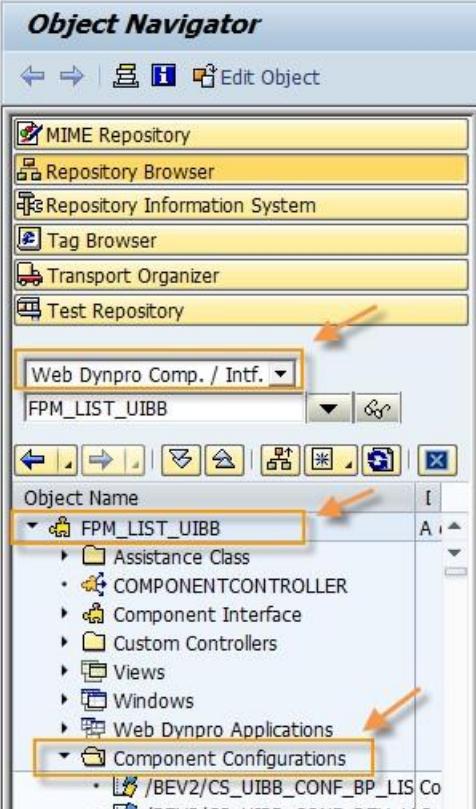
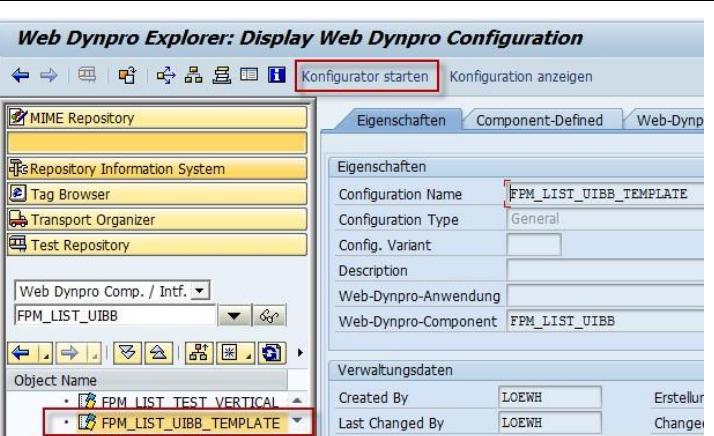
- Choose the **Start Configurator** pushbutton.

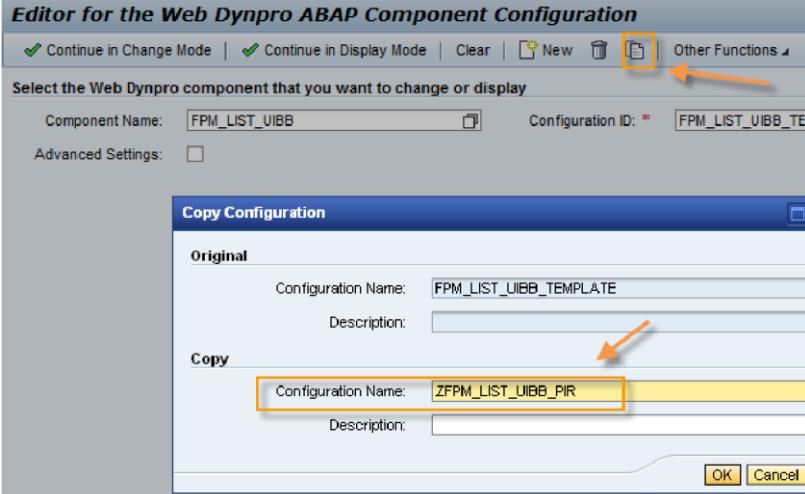
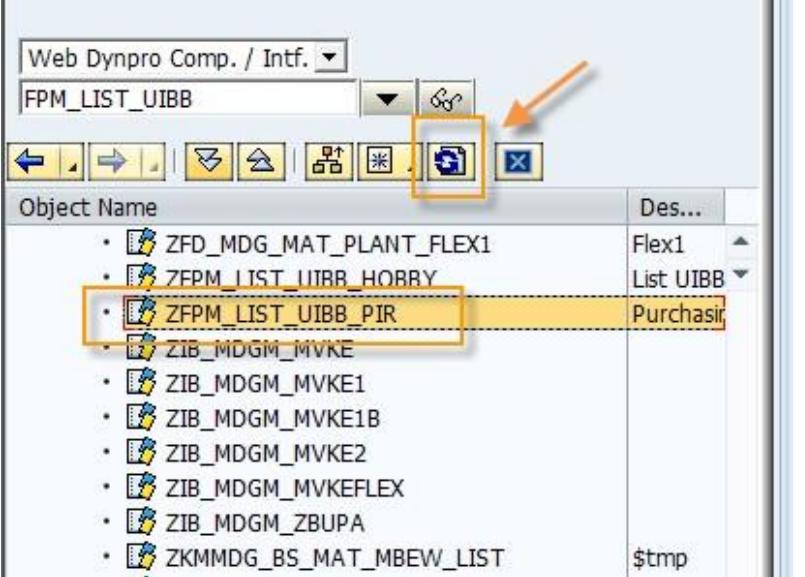


3. Copy the existing configuration to ZZBS_SP_OVP .	
4. Refresh the list of configurations and look for your copy. It should be there.	

2.2.4 Create List UIBB

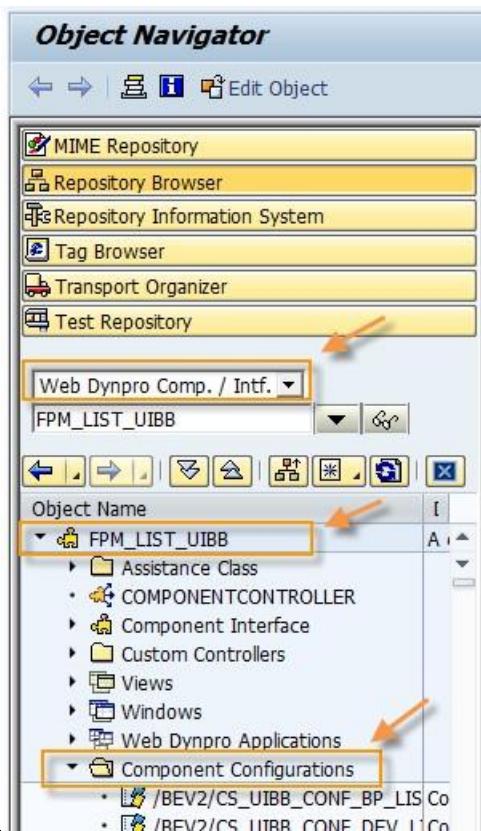
Copy List UIBB

<p>1. Start transaction SE80. Navigate to the Component Configuration as shown.</p>	 <p>The screenshot shows the SAP Object Navigator interface. The left pane lists various repository components: MIME Repository, Repository Browser, Repository Information System, Tag Browser, Transport Organizer, and Test Repository. The right pane shows the 'Web Dynpro Comp. / Intf.' dropdown set to 'FPM_LIST_UIBB'. Under 'Object Name', the 'FPM_LIST_UIBB' node is expanded, revealing its sub-components: Assistance Class, COMPONENTCONTROLLER, Component Interface, Custom Controllers, Views, Windows, and Web Dynpro Applications. The 'Component Configurations' folder under Windows is also expanded, showing two entries: '/BEV2/CS_UIBB_CONF_BP_LIS Co' and '/BEV2/CS_UIBB_CONF_DEV_L1Co'. Arrows point from the 'Object Navigator' title bar to the 'Object Navigator' label in the top-left of the right pane, and from the 'Object Name' label to the expanded 'Component Configurations' node.</p>
<p>2. Locate the FPM_LIST_UIBB_TEMPLATE in the list of component configurations. Choose the Start Configurator pushbutton.</p>	 <p>The screenshot shows the SAP Web Dynpro Explorer interface. The left pane displays the same repository components as the Object Navigator. The right pane is titled 'Web Dynpro Explorer: Display Web Dynpro Configuration'. It shows the properties for a configuration named 'FPM_LIST_UIBB_TEMPLATE'. The properties tab is selected, displaying the following details:</p> <ul style="list-style-type: none"> Eigenschaften tab: Configuration Name is 'FPM_LIST_UIBB_TEMPLATE', Configuration Type is 'General', and Web-Dynpro-Component is 'FPM_LIST_UIBB'. Component-Defined tab: Contains fields for Description, Web-Dynpro-Anwendung, and Verwaltungsdaten (Created By: LOEWH, Last Changed By: LOEWH). Web-Dynp tab: Contains fields for Erstellur and Change. <p>The 'FPM_LIST_UIBB_TEMPLATE' entry is highlighted in the list of configurations in the left pane.</p>

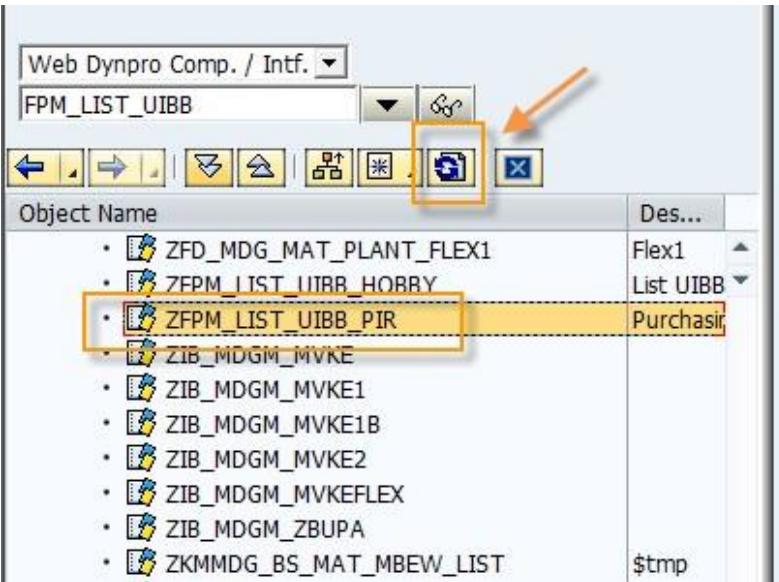
3. Copy the existing configuration to ZFPM_LIST_UIBB_PIR .																							
4. Refresh the list of configurations and look for your copy. It should be there.	 <table border="1" data-bbox="690 887 1405 1277"> <thead> <tr> <th>Object Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>ZFD_MDG_MAT_PLANT_FLEX</td> <td>Flex1</td> </tr> <tr> <td>ZFPM_LIST_UITRR_HOBRY</td> <td>List UBB</td> </tr> <tr> <td>ZFPM_LIST_UIBB_PIR</td> <td>Purchasing</td> </tr> <tr> <td>ZIB_MDGM_MVKE</td> <td></td> </tr> <tr> <td>ZIB_MDGM_MVKE1</td> <td></td> </tr> <tr> <td>ZIB_MDGM_MVKE1B</td> <td></td> </tr> <tr> <td>ZIB_MDGM_MVKE2</td> <td></td> </tr> <tr> <td>ZIB_MDGM_MVKEFLEX</td> <td></td> </tr> <tr> <td>ZIB_MDGM_ZBUPA</td> <td></td> </tr> <tr> <td>ZKMMDG_BS_MAT_MBEW_LIST</td> <td>\$tmp</td> </tr> </tbody> </table>	Object Name	Description	ZFD_MDG_MAT_PLANT_FLEX	Flex1	ZFPM_LIST_UITRR_HOBRY	List UBB	ZFPM_LIST_UIBB_PIR	Purchasing	ZIB_MDGM_MVKE		ZIB_MDGM_MVKE1		ZIB_MDGM_MVKE1B		ZIB_MDGM_MVKE2		ZIB_MDGM_MVKEFLEX		ZIB_MDGM_ZBUPA		ZKMMDG_BS_MAT_MBEW_LIST	\$tmp
Object Name	Description																						
ZFD_MDG_MAT_PLANT_FLEX	Flex1																						
ZFPM_LIST_UITRR_HOBRY	List UBB																						
ZFPM_LIST_UIBB_PIR	Purchasing																						
ZIB_MDGM_MVKE																							
ZIB_MDGM_MVKE1																							
ZIB_MDGM_MVKE1B																							
ZIB_MDGM_MVKE2																							
ZIB_MDGM_MVKEFLEX																							
ZIB_MDGM_ZBUPA																							
ZKMMDG_BS_MAT_MBEW_LIST	\$tmp																						

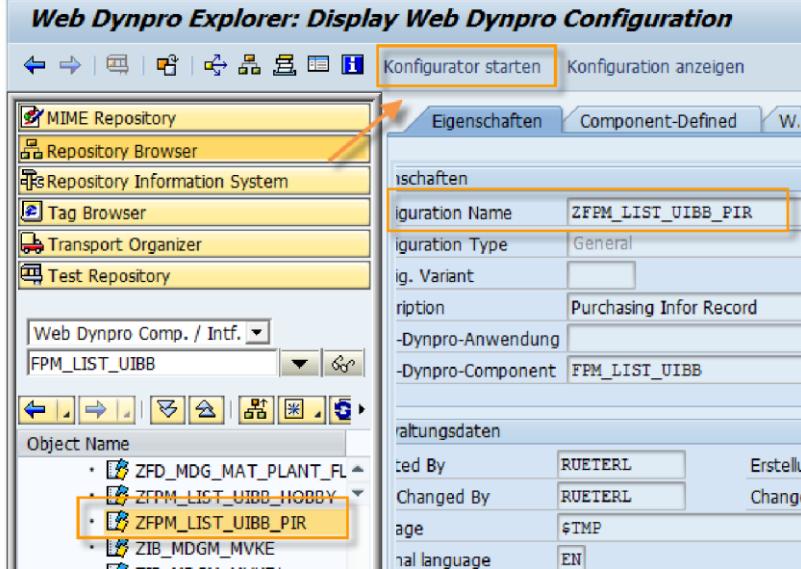
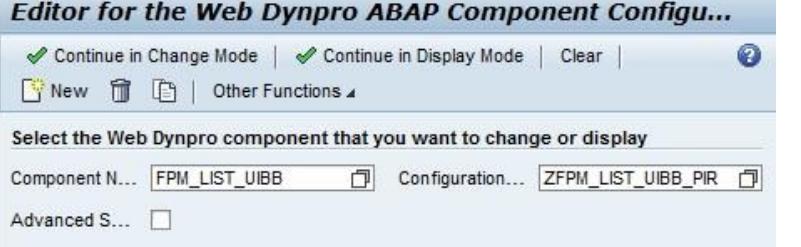
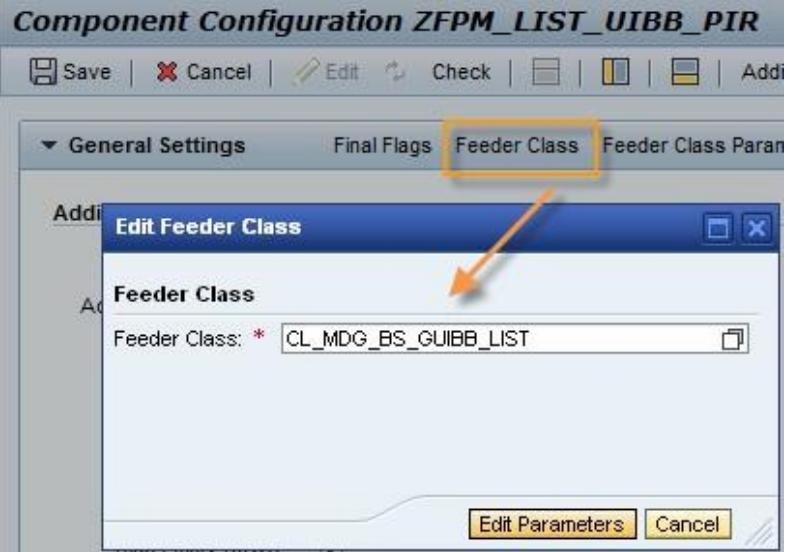
Extend UIBB List Component

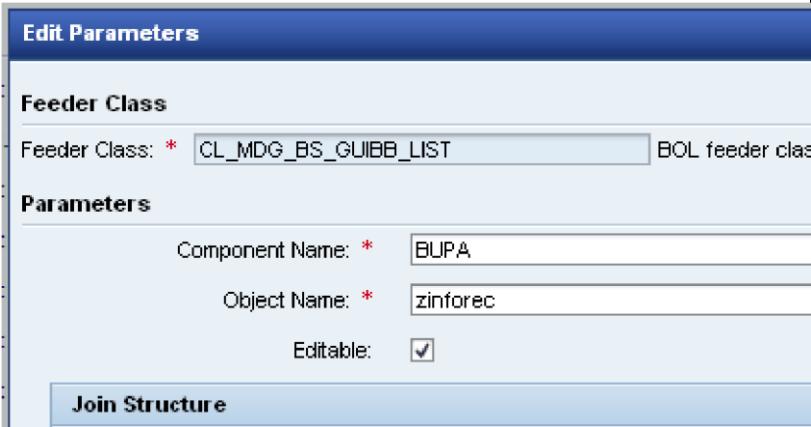
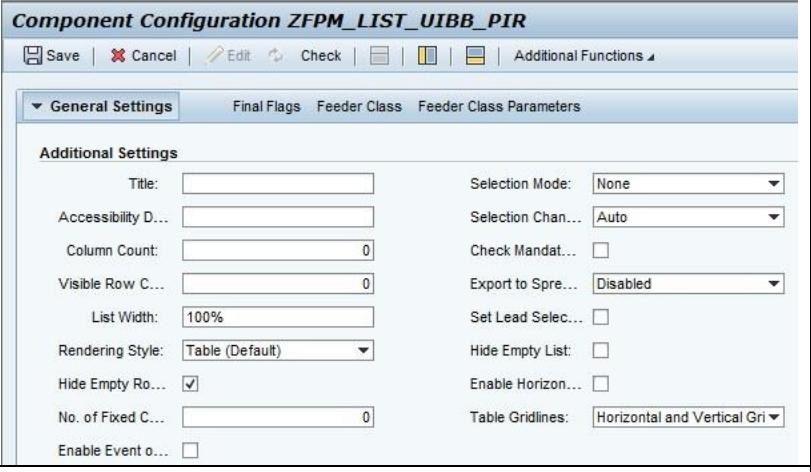
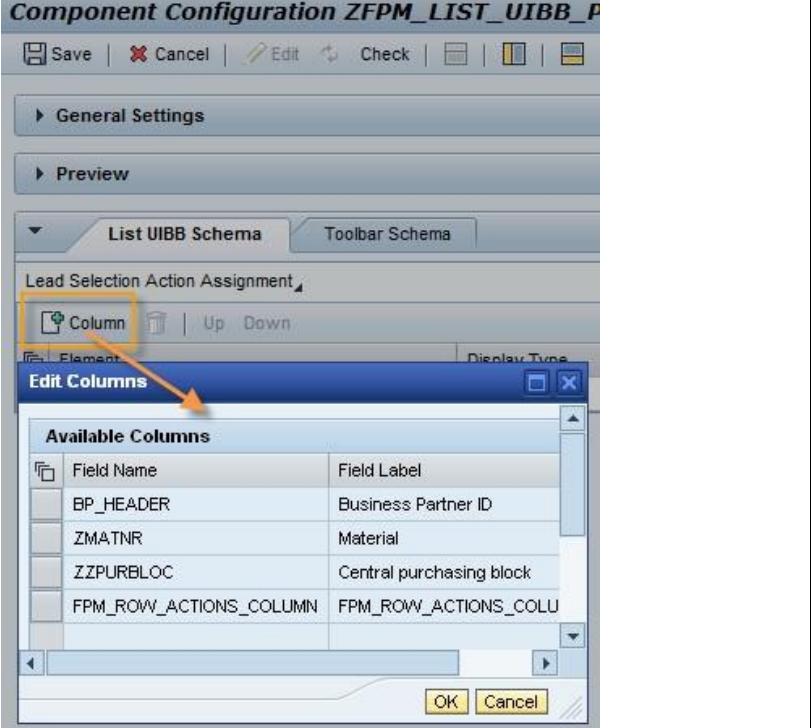
1. Start transaction **SE80**.
Navigate to the **Component Configuration** as shown.

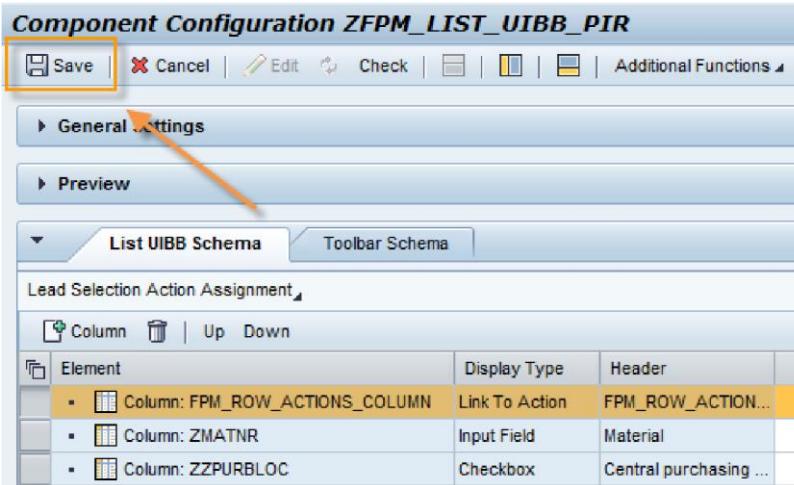
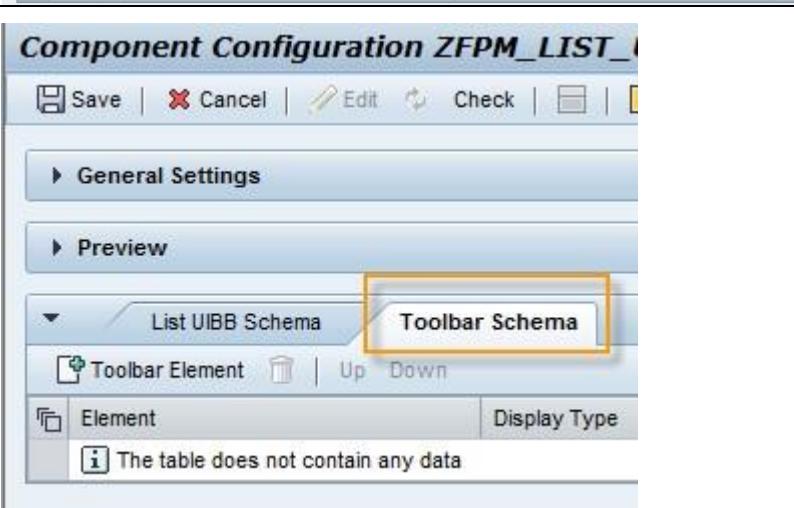


2. Locate your own **ZFPM_LIST_UIBB_PIR** configuration in the list of component configurations.

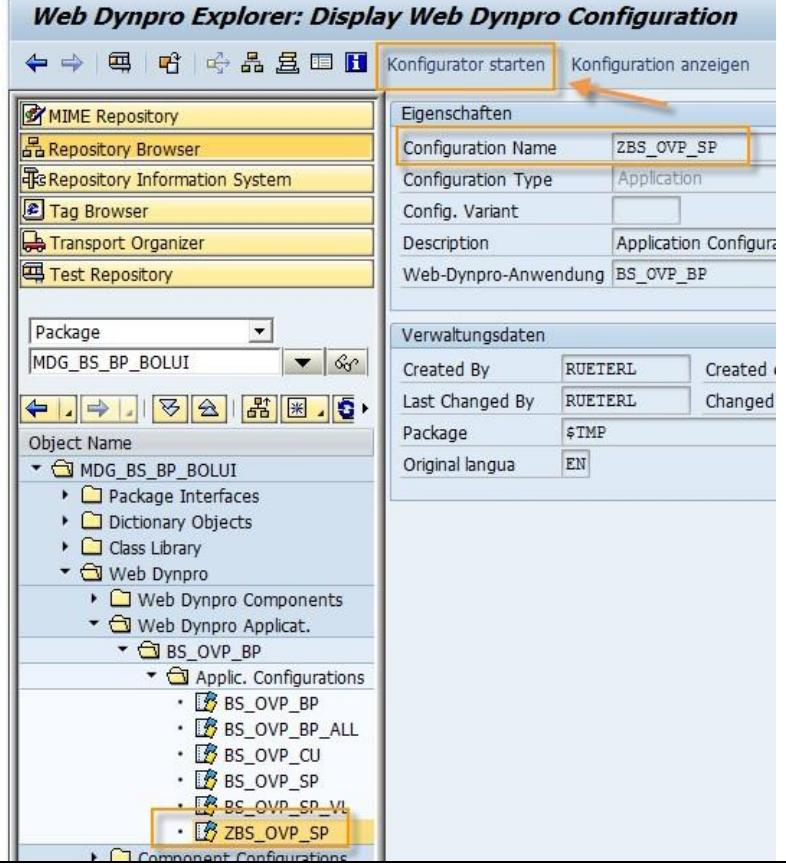
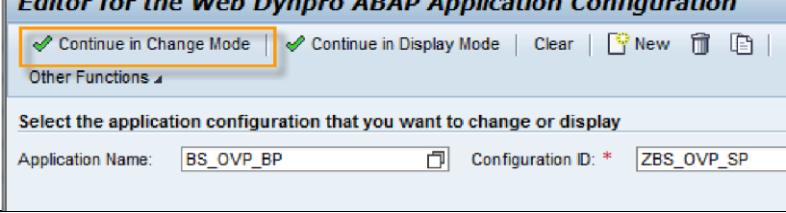
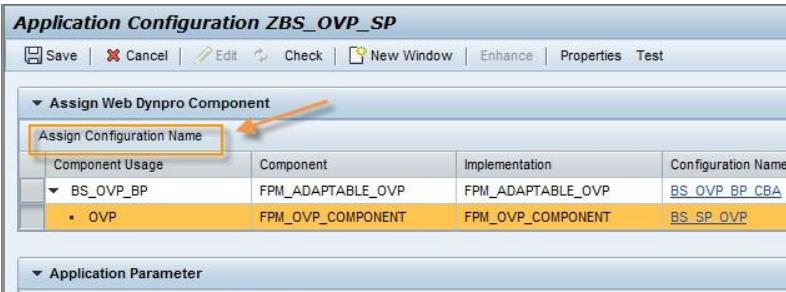


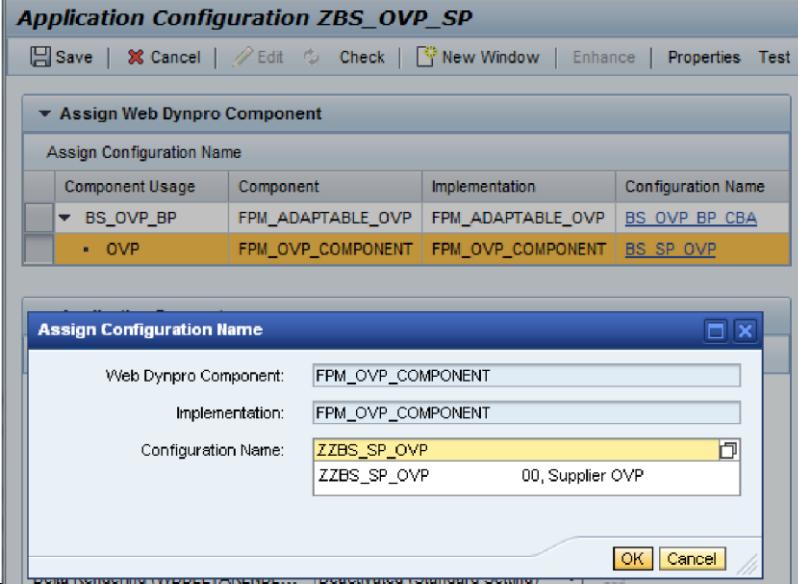
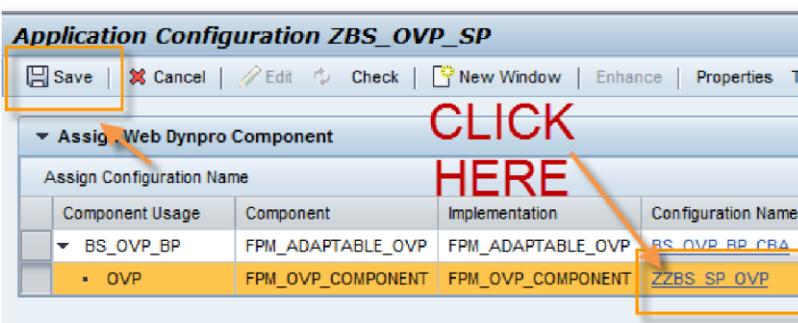
3. Choose the Start Configurator pushbutton.	 <p>Web Dynpro Explorer: Display Web Dynpro Configuration</p> <p>Konfigurator starten Konfiguration anzeigen</p> <p>Eigenschaften Component-Defined W.</p> <p>Properties</p> <p>Configuration Name: ZFPM_LIST_UIBB_PIR</p> <p>Configuration Type: General</p> <p>Config. Variant:</p> <p>Description: Purchasing Infor Record</p> <p>-Dynpro-Anwendung:</p> <p>-Dynpro-Component: FPM_LIST_UIBB</p> <p>Maintenance Data</p> <p>Created By: RUETERL Erstellt am: 2023-09-11 Change Date: 2023-09-11</p> <p>Changed By: RUETERL Change Date: 2023-09-11</p> <p>Language: EN</p> <p>Object Name:</p> <ul style="list-style-type: none"> ZFD_MDG_MAT_PLANT_FL ZFPM_LIST_UIBB_HOBBY ZFPM_LIST_UIBB_PIR (highlighted) ZIB_MDGM_MVKE ZIB_HOBBY_MVKE
4. Choose the Continue in Change Mode pushbutton.	 <p>Editor for the Web Dynpro ABAP Component Configuration</p> <p>Continue in Change Mode Continue in Display Mode Clear ?</p> <p>New Other Functions</p> <p>Select the Web Dynpro component that you want to change or display</p> <p>Component N... FPM_LIST_UIBB Configuration... ZFPM_LIST_UIBB_PIR Advanced S... </p>
5. Choose the Feeder Class pushbutton and enter the feeder class as shown.	 <p>Component Configuration ZFPM_LIST_UIBB_PIR</p> <p>Save Cancel Edit Check Add</p> <p>General Settings Final Flags Feeder Class Feeder Class Parameters</p> <p>Add Edit Feeder Class</p> <p>Feeder Class: * CL_MDG_BS_GUIBB_LIST</p> <p>Edit Parameters Cancel</p>

6. Enter the feeder class parameters as shown.	
7. Enter the general settings of the UIBB as shown.	
8. Add your fields as table columns as shown.	

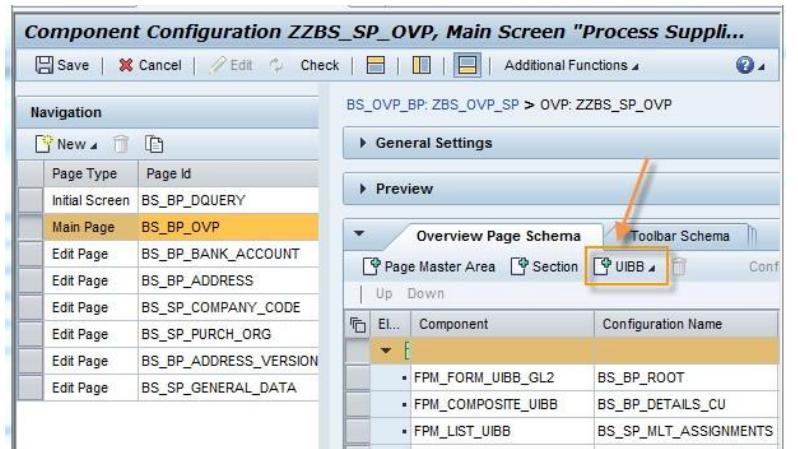
9. Save your changes.	 <p>Component Configuration ZFPM_LIST_UIBB_PIR</p> <p>Save Cancel Edit Check Additional Functions</p> <p>General Settings</p> <p>Preview</p> <p>List UIBB Schema Toolbar Schema</p> <p>Lead Selection Action Assignment</p> <table border="1"> <thead> <tr> <th>Element</th> <th>Display Type</th> <th>Header</th> </tr> </thead> <tbody> <tr> <td>Column: FPM_ROW_ACTIONS_COLUMN</td> <td>Link To Action</td> <td>FPM_ROW_ACTION...</td> </tr> <tr> <td>Column: ZMATNR</td> <td>Input Field</td> <td>Material</td> </tr> <tr> <td>Column: ZZPURBLOC</td> <td>Checkbox</td> <td>Central purchasing ...</td> </tr> </tbody> </table>	Element	Display Type	Header	Column: FPM_ROW_ACTIONS_COLUMN	Link To Action	FPM_ROW_ACTION...	Column: ZMATNR	Input Field	Material	Column: ZZPURBLOC	Checkbox	Central purchasing ...
Element	Display Type	Header											
Column: FPM_ROW_ACTIONS_COLUMN	Link To Action	FPM_ROW_ACTION...											
Column: ZMATNR	Input Field	Material											
Column: ZZPURBLOC	Checkbox	Central purchasing ...											
10. Nothing needs to be added to the Toolbar Schema tab page.	 <p>Component Configuration ZFPM_LIST_UIBB_PIR</p> <p>Save Cancel Edit Check Additional Functions</p> <p>General Settings</p> <p>Preview</p> <p>List UIBB Schema Toolbar Schema</p> <p>Toolbar Element Up Down</p> <table border="1"> <thead> <tr> <th>Element</th> <th>Display Type</th> </tr> </thead> <tbody> <tr> <td>The table does not contain any data</td> <td></td> </tr> </tbody> </table>	Element	Display Type	The table does not contain any data									
Element	Display Type												
The table does not contain any data													

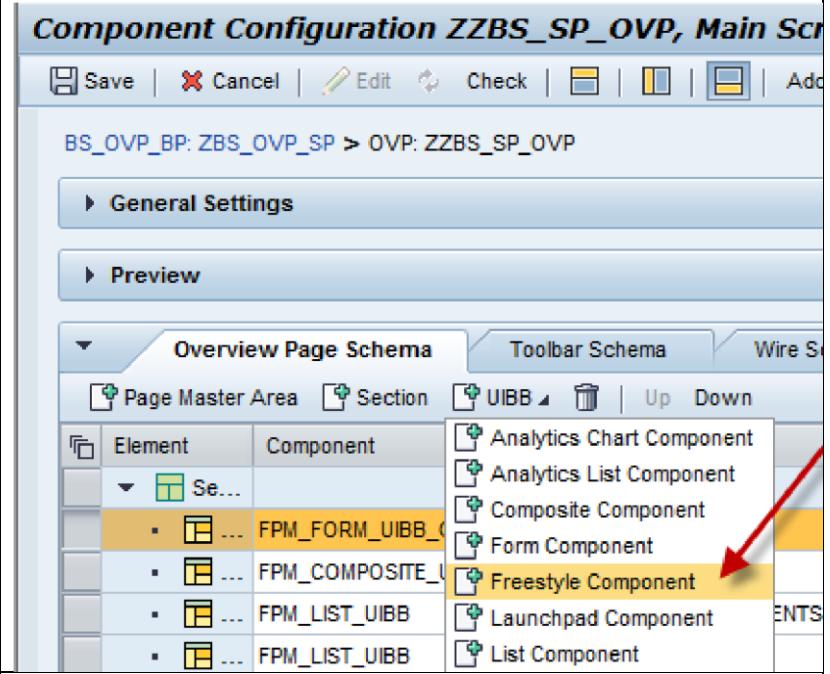
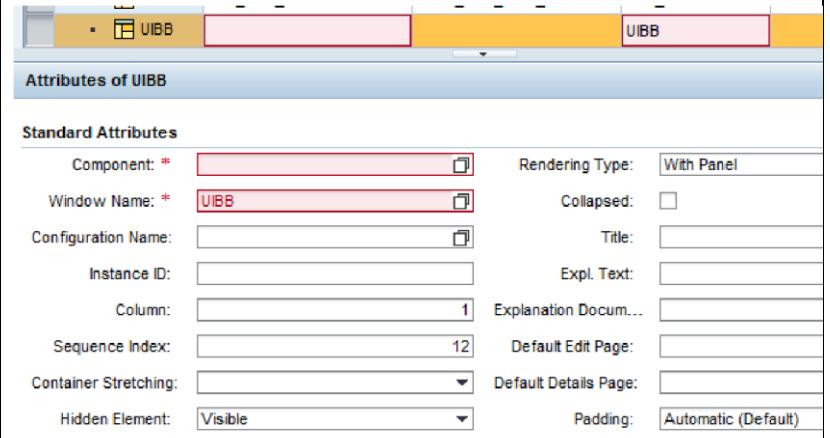
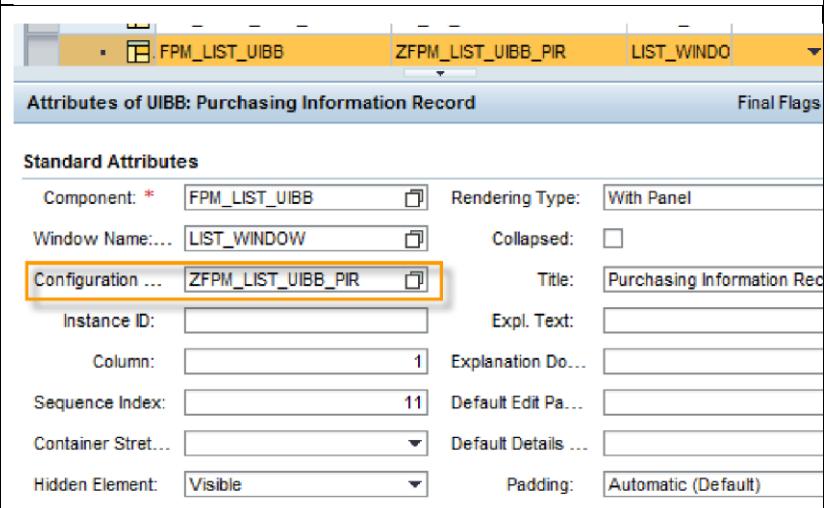
2.2.5 Replace OVP Component in Application Configuration for Supplier

<p>1. Start transaction SE80. Navigate to your Application Configuration as shown.</p>	 <p>The screenshot shows the 'Web Dynpro Explorer: Display Web Dynpro Configuration' interface. The left pane displays a tree view of objects under 'MDG_BS_BP_BOLUI' package, including 'Package Interfaces', 'Dictionary Objects', 'Class Library', 'Web Dynpro' (which contains 'Web Dynpro Components' and 'Web Dynpro Applications'), and 'Applic. Configurations' (which contains 'BS_OVP_BP', 'BS_OVP_BP_ALL', 'BS_OVP_CU', 'BS_OVP_SP', 'BS_OVP_SP_VL', and 'ZBS_OVP_SP'). The right pane shows configuration details: Configuration Name is 'ZBS_OVP_SP', Configuration Type is 'Application', and Description is 'Application Configuration'. Buttons at the top right include 'Konfigurator starten' and 'Konfiguration anzeigen'.</p>
<p>2. Choose the Continue in Change Mode pushbutton.</p>	 <p>The screenshot shows the 'Editor for the Web Dynpro ABAP Application Configuration' interface. It features a toolbar with 'Continue in Change Mode' (selected), 'Continue in Display Mode', 'Clear', and other functions. Below is a search bar for 'Application Name: BS_OVP_BP' and 'Configuration ID: * ZBS_OVP_SP'.</p>
<p>3. Select the row starting with ovp and choose the pushbutton Assign Configuration Name.</p>	 <p>The screenshot shows the 'Application Configuration ZBS_OVP_SP' dialog. It has tabs for 'Assign Web Dynpro Component' (selected) and 'Application Parameter'. In the 'Assign Web Dynpro Component' tab, there is a table with columns: Component Usage, Component, Implementation, and Configuration Name. A row for 'OVP' is selected, showing 'FPM_OVP_COMPONENT' in the Component column and 'BS_SP_OVP' in the Configuration Name column. An arrow points to the 'Assign Configuration Name' button in the toolbar above the table.</p>

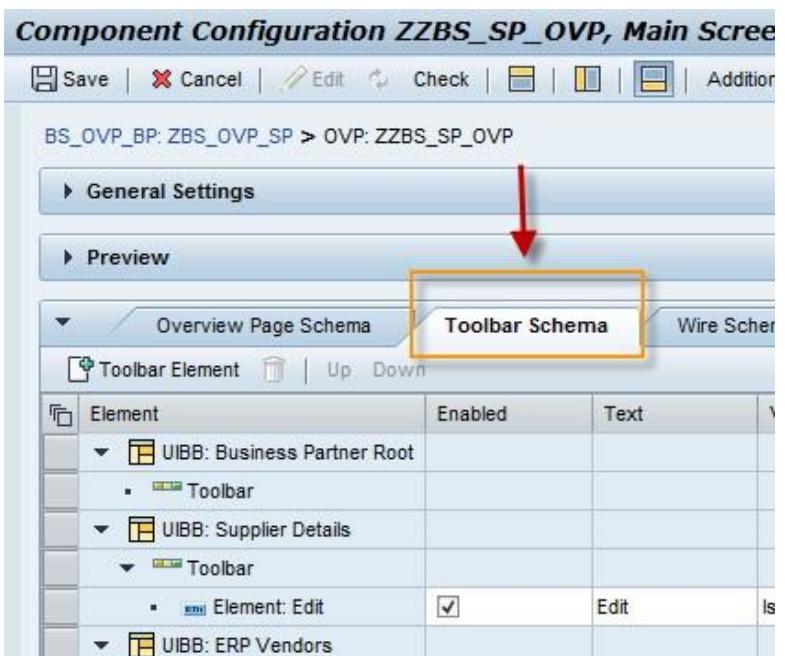
4. Change the configuration name as shown.	
5. Save your changes. Click on the link ZZBS_SP_OVP to edit the configuration.	

2.2.6 Add Custom List UIBB to Supplier OVP

1. Open the pushbutton UIBB menu, as shown.	
--	--

2. Select Freestyle Component from the drop down as shown.	
3. The empty property list appears.	
4. Enter the attribute values as shown.	

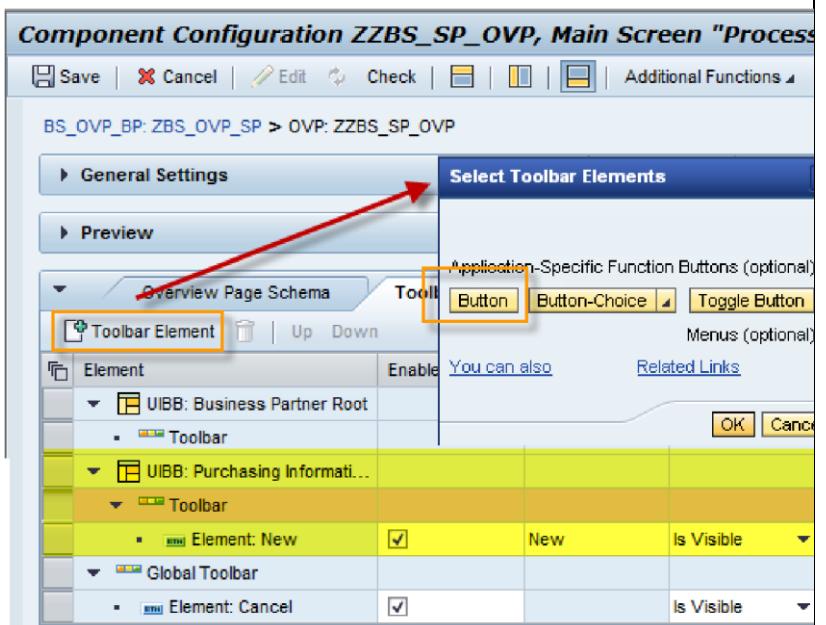
5. Choose the **Toolbar Schema** tab.



6. Choose the **Toolbar Element** pushbutton.

Choose the **Button** pushbutton.

Change the name of the pushbutton to **New** and assign the CREATE event to the pushbutton.



7. Enter the attribute values as shown.	
8. You need to add an Event Parameter in the table. If you do not enter this event parameter, you will later have problems in the UI with values disappearing after entry. Click on the Add Parameter pushbutton and add the following parameter: Parameter Name: DEFERRED_SENDING Parameter Value: x	
9. Change to the Wire Schema tab. Choose the Wire pushbutton.	

10. Enter the attribute values as shown.

Standard Attributes

Component: FPM_LIST_UIBB

Configuration Name:

ZFPM_LIST_UIBB_PIR

Instance ID:

Source Component:

FPM_FORM_UIBB_GL2

Source Config Name:

BS_BP_ROOT

Srce Inst. ID:

Port Type: Lead Selection

Port Identifier: STANDARD

Connector Class: *

CL_BS_BP_CONNECTOR_BOL_REL

Connector Parameters

Relation Name: zbp2inforec

Creation Mode: Creation with Default Values

Standard Attributes	
Component: *	FPM_LIST_UIBB
Configuration Na...	ZFPM_LIST_UIBB_PIR
Instance ID:	
Source Compon...	FPM_FORM_UIBB_GL2
Source Config N...	BS_BP_ROOT
Port Type:	Lead Selection
Port Identifier:	STANDARD
Connector Class...:	CL_BS_BP_CONNECTOR_BOL_REL

Connector Parameters	
Relation Name:	zbp2inforec
Creation Mode:	Creation with Default Values

2.3 Testing the Configuration

To test your configuration, start the MDG Supplier UI using the following URL and replace the parameter value **WDCONFIGURATIONID** with the name of your copy of the standard configuration.

URL: http://<host>:<port>/sap/bc/webdynpro/sap/bs_ovp_bp?sap-client=405&saplanguag=EN&WDCONFIGURATIONID=ZBS_OVP_SP

Organization: \$118, (no description available)

Save Submit | Check

ERP Vendors

Replication Status

Actions	*ERP Vendor	Reason	Standard
[i] The table does not contain any data			

Roles

Addresses

Address Usages

Bank Accounts

Actions	*ID	Bank Key	Ban...	Bank Name	Bank Account	Account Name	Vali...	Valid...
[i] The table does not contain any data								

Identification Numbers

Tax Numbers

Industries

Purchasing Information Record

Actions	*Material	Central purchasing block
[i] The table does not contain any data		

Done

Trusted sites | Protected Mode: Off

100%

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](#)
- Learn more: [Latest Release](#) | [Webinars](#) | [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.

Note	Description
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
3194967	MDG Customer Connection 2021 for S/4HANA 2022
3043582	MDG Customer Connection 2020
2479869	Usage of Lean Classification with SAP Master Data Governance

1619534	How to Create, Enhance and Adapt FPM Applications
1637249	MDG: Information for efficient message processing
2105467	MDG Performance
2561461	Scope of support for SAP Master Data Governance (MDG)