PUBLIC

How To… Develop a Custom Object with SAP MDG (Master Data Governance)

Implementing a custom object master data domain with SAP MDG incl. Data Model, Process Model and User Interface

Applicable Releases: All

Version 2.0

May 2019

THE BEST RUN SAP®

DOCUMENT HISTORY

Document Version	Description
1.00	First official release of this guide on MDG6.1 EHP6
2.00	Major updated version for MDG 9.1 EHP8 and S/4HANA 1709. Usage of BOL/genIL in MDG Custom objects. More complex data model.

www.sap.com/contactsap

THE BEST RUN SAP®

Table of Contents

INTRO

Author Bio

Steffen Ulmer is an SAP Principal Consultant who works at SAP (Schweiz) AG. In this role, he has been a trusted advisor for large enterprises on topics and products like: SAP Master Data Governance, Central Governance, Consolidation and Mass Processing, Master Data Strategy and Architecture.

Summary

Many companies want to manage custom objects in a central Master data system to be able to harmonize this information across their landscape. Custom objects can be individual defined objects such as sites or plants. Custom objects are typically less complex master data objects with a small and simple data model. They are often used as reference data in major objects such as material, suppliers and customers.
This How-To Guide describes the necessary steps to implement a custom object in SAP MDG, central governance. The guide is based on an object called “SITE” and includes the following steps during the implementation phase:

- Create a new Data Model
- Define a new Business Object and Activity
- Create a custom User Interface
- Process Modeling for logical action CREATE
- Test of the custom object

At the end of the document, you will find further information about potential next steps to further enhance the custom object application by a more complex data model and additional scenarios such as change processes or transforming reference data into customizing.
The audience of this document can be customers, technology consultants and architects.

About this Document

SAP Master Data Governance is a Master Data Management solution within the Business Suite and SAP S/4HANA, and is integrated into the ABAP Application Server of SAP.
I really think that lots of you will have similar requirements in an MDG-based environment, and lots of you want to develop custom objects and processes which are not (and cannot be) delivered within the standard content.
This document will describe how to develop a custom object in SAP MDG. It will show you, in some very basic steps, what you need to do to:

- Get your own data model into the system
- Configure your own User Interface with the SAP FPM (Floor Plan Manager)
- Reuse a preconfigured Business Workflow within your own Change Request Type
- Test your new custom object via a Change Request

Prerequisites

For this How-To Guide you should have access to a sandbox of your MDG Application Server with appropriate access rights.
The ERP System must run on EhP (Enhancement Package) 8 and at least version SAP MDG 9.0 or on SAP S/4HANA 1709.
You need some basic understanding of the following topics:

- SAP MDG
- SAP Business Workflow
- SAP Floorplan Manager (FPM)

Development skills are not required since not one single line of code will be written. If you want to save the configuration into an ABAP transport request, you need a workbench and customizing request/tasks.

Depending on your experience levels, you will need 2-4 hours to complete the tasks in this guide.

Scenario

Let`s assume that you have already successfully implemented the SAP MDG standard objects such as Material, Suppliers, Financials and Customers. You are replicating this data to several ERP and Non-ERP Systems for usage within transactional matters.
Your business department is generally happy with the solution, but they have an issue with some of the reference data. Reference data is information which is used within master data objects itself, or in transactions or programs as supporting information. In SAP, such data is typically displayed in F4 helps or drop-down value lists. Examples of such reference data are:

- List of plants which is used to assign a material or products to a physical facility
- List of countries which is used within many transactions and referenced in many data models.

Your business users identified that these lists of valid reference data are not harmonized within the landscape. This gap causes issues regarding data quality which ends up in inefficient business processes. A concrete example is the information object “Site” which is a valid list of geographical places where the company can have plants, unloading points or just generally assets.

High-Level Requirements

The business requests the following:

1. A central system in which the valid list of values for sites are managed
2. A governance process to be able to carefully add (or change) site objects
3. A web-based application to enter and govern the data in a User Interface
4. A search application to find existing sites

Data Model

In this How-To Guide the SAP MDG flex mode is used.

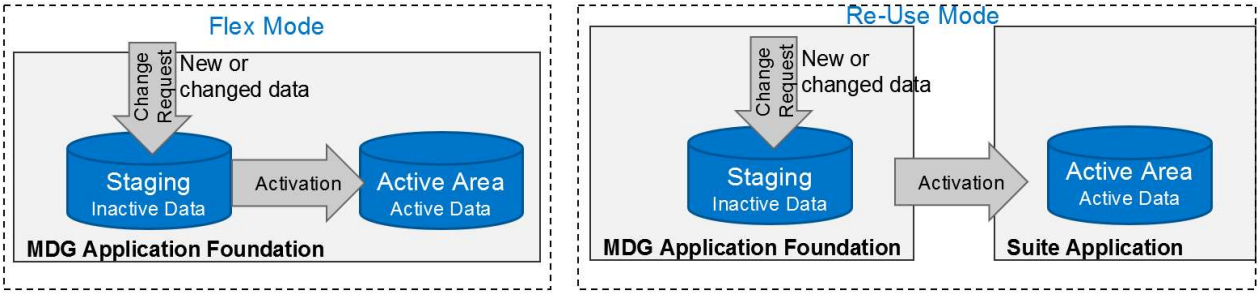
Handling of Active And Inactive Data
for Custom Objects

MDG separates

- **Active area** – Holds data that is ready to be used by applications
- **Staging** – Holds the data that is not yet approved, currently part of a change request

For optimized integration and flexibility MDG allows two modes for the *Active Area*

- **Re-Use active area (re-use mode)** – Existing structures (i.e. data base tables) of applications are used. For example, MDG for material makes use of the MARA table in ECC. Re-Use Class needed. Used for MDG-S, MDG-C, MDG-M. Can be used for Custom Objects (starting with EhP6)
- **Generated active area (flex mode)** – Tables as defined in the MDG data model are used to store active data. Used for MDG-F. Has to be used for Custom Objects in EhP6.



Governance Process

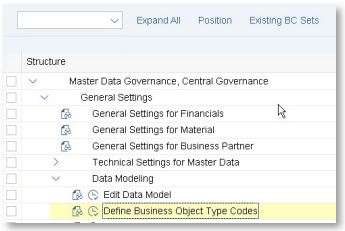
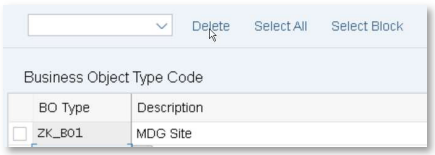
The governance process is also very simple and includes just a 2-step approval. The following graphic shows a high-level view of the governance process:

This guide uses the SAP MDG Rule-Based Workflow template to model this process.

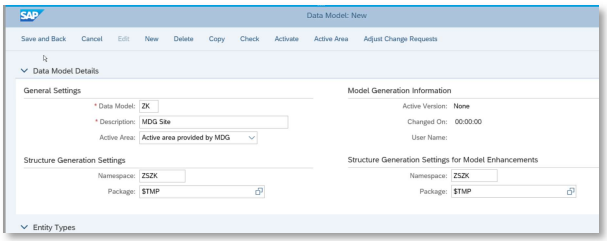
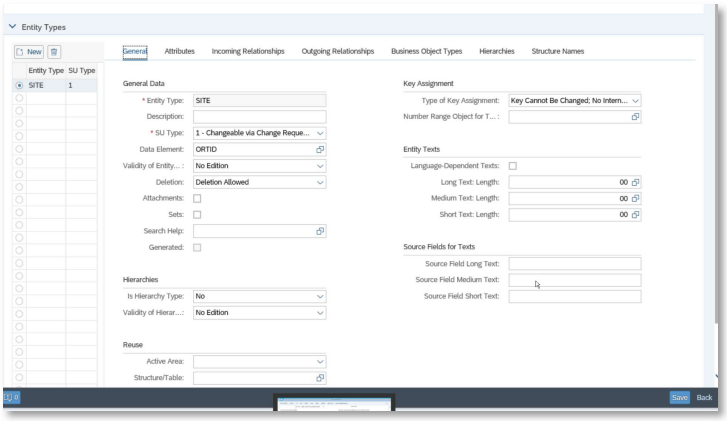
STEP-BY-STEP GUIDE

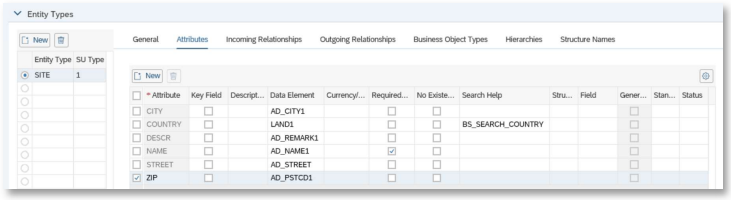
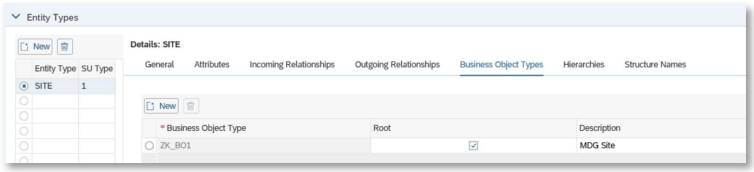
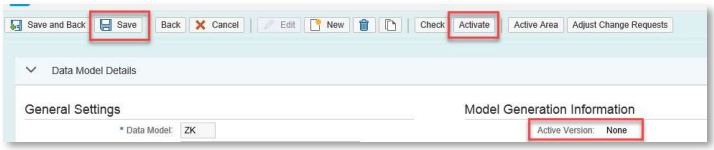
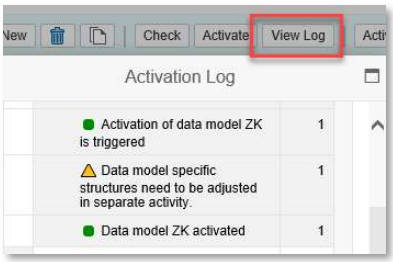
Data Modeling

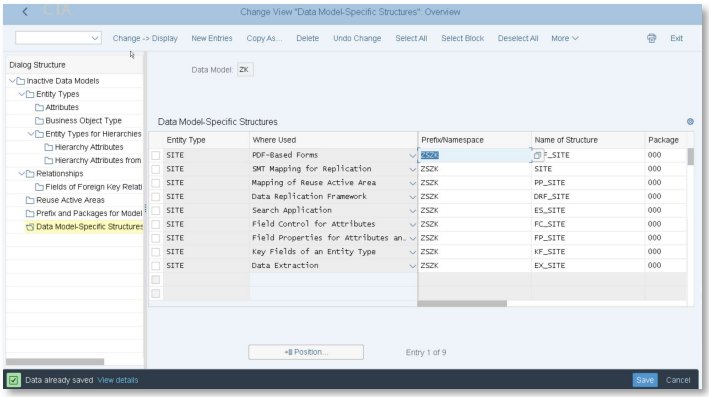
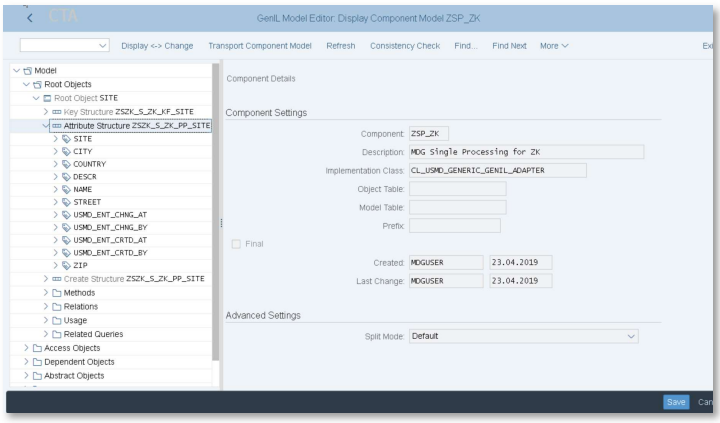
Create Business Object Type Code (OTC)

Logon with SAP GUI and start transaction MDGIMG. Navigate to *General Settings → Data Modeling → Define Business Object Type Codes*	
Use *New Entries* to create BO Type ZK_BO1	
Save your changes	

Create New Data Model

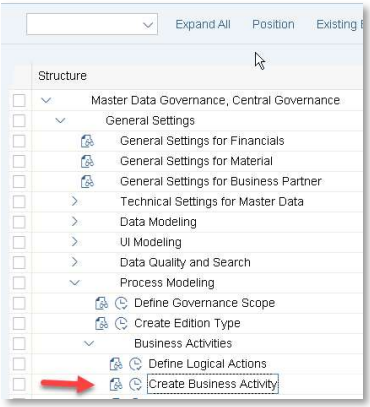
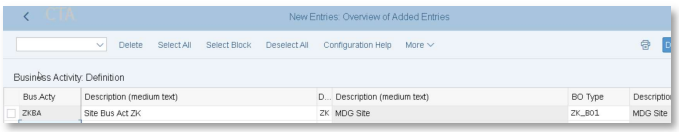
Navigate to *Configuration Workbench* (alternatively you can use the old "*Edit Data Model*")	
Use *New Entries* to create a new Data Model with the following: **Data Model: ZK** **Descr.: MDG Site** **Active Area: provided by MDG** **Namespace: ZSZK** **Package: $TMP**	
Click *New* in the *Entity Types* section Add **SITE** Use the values as shown on the right: **Storage: Type 1** **Data Element: ORTID**	

Click on *Attributes* tab and add *New* attributes for the domain: (Attribute) (Data Element) **CITY** **AD_CITY1** **COUNTRY** **LAND1** **DESCR** **AD_REMARK1** **NAME** **AD_NAME1** **STREET** **AD_STREET** **ZIP** **AD_PSTCD1** Select the *Required Entry* checkbox for the NAME attribute	
Click *Business Object Types* and assign the BO Type **ZK_BO1** Select the *Root* checkbox	
Save the Data Model	
Activate the Data Model System will show a success message	This will take several seconds. The system generates the active area and all dependent ABAP Structures  Info: The system generates many structures and also the genIL model (for single processing, multi-record processing and even hierarchy processing)
View Log	
Verify Active Version Go back into the data model and check the *Model Generation Information*	Active Version should be *Same*  

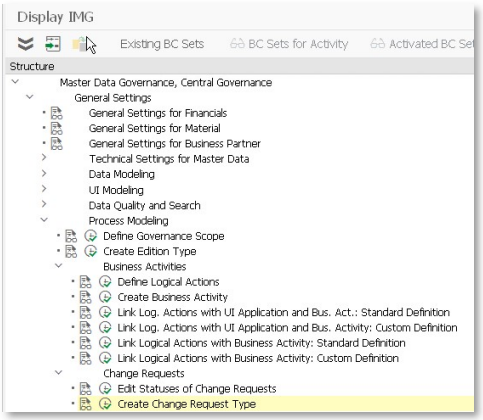
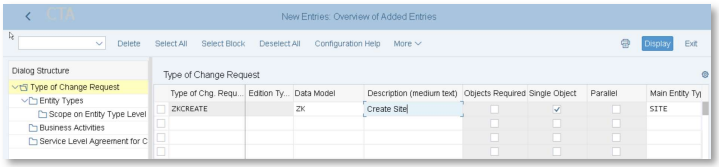
[OPTIONAL] **Verify the structure and the generated tables** Run report USMD_DATA_MODEL via SE37/38	
[OPTIONAL] Verify that structures have been created. If they are not created, regenerate the data model using classic IMG “Data Model” (repeat activation) Use se80 to review	
[OPTIONAL] Call transaction genil_model_browser to verify that the genIL structures have been created Component: ZSP_ZK	 

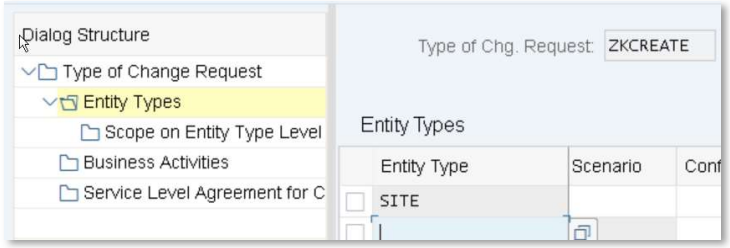
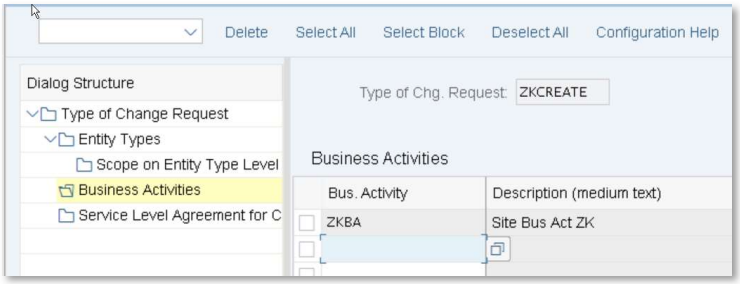
Workflow and Process Modeling

Create New Bus. Activity

Start MDGIMG and navigate to *Create Business Activity*	
Use *New Entries* to create a business activity as follows: **Bus. Activity: ZKBA** **Descr: Site Bus Act ZK** **Data Model: ZK** **BO Type: ZK_BO1** **Log. Action: CREATE**	
Save your changes	

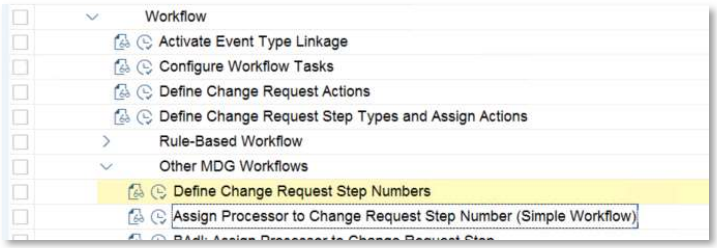
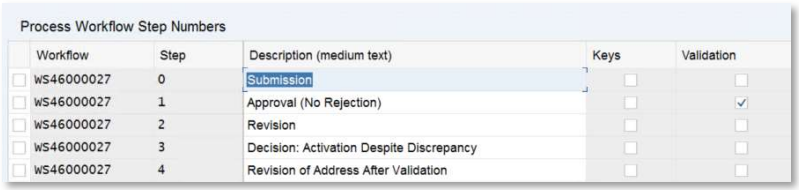
Create New CR Type

Start MDGIMG and navigate to *Create Change Request Type*	
Use *New Entries* to create a new CR as shown. **Type of CR: ZKCREATE** **Data Model: ZK** Don't forget to mark the **checkbox for Single Object processing** **Main Entity Type: SITE** Use WF Template **WS46000027** which has the 2-step approval flow	

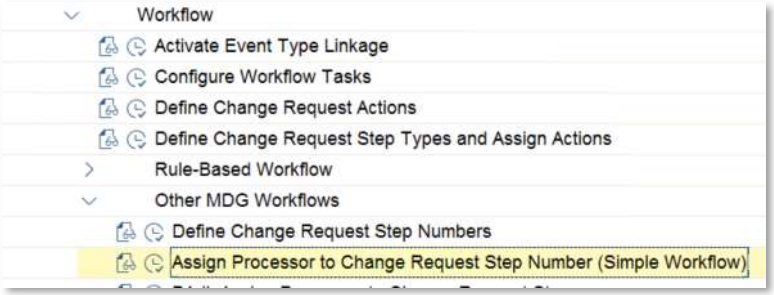
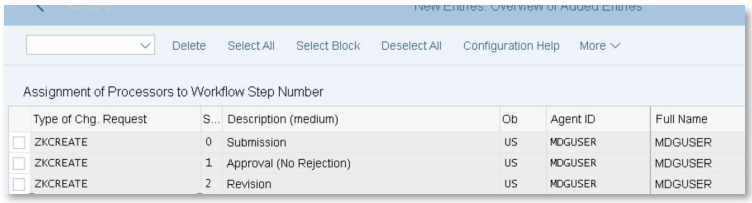
Assign the **Entity Type SITE** to the CR Type	
Assign **Bus. Activity ZKBA to the CR Type**	
Save your settings	

Configure/Create Workflow Model

Revise Steps for Standard Workflow

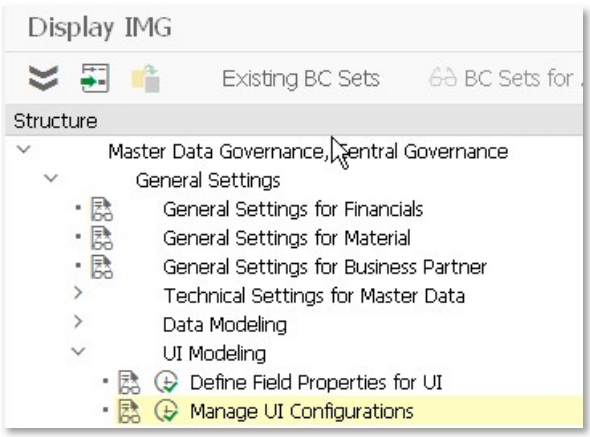
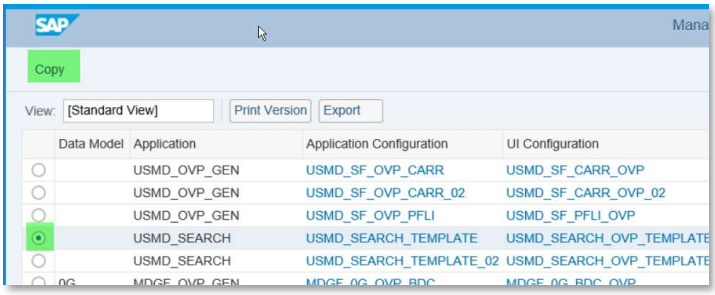
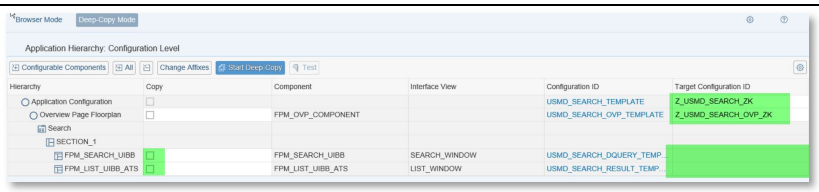
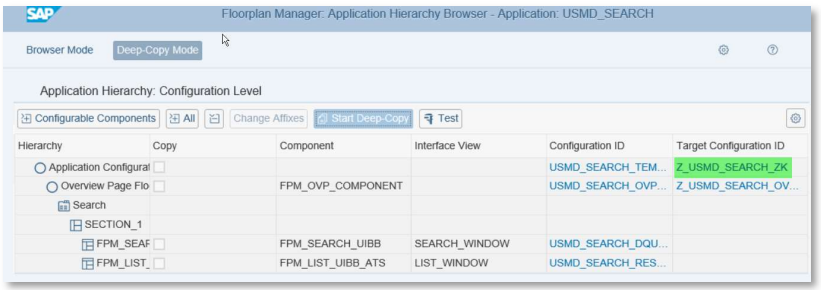
Navigate to *Define Change Request Step Numbers* and look for WS46000027	
Verify that steps 0-4 are maintained, otherwise add them using the descriptions shown on the right	

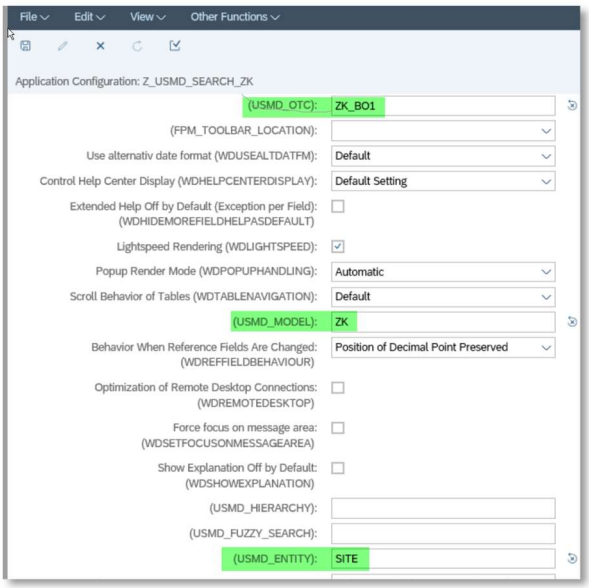
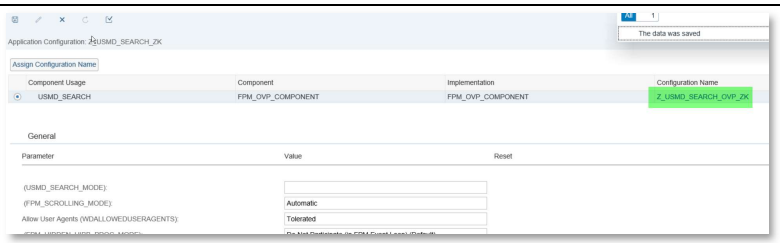
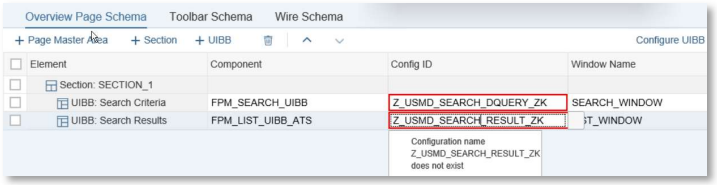
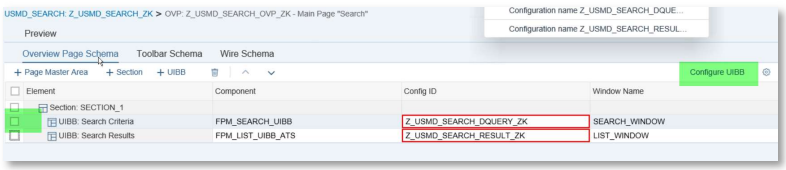
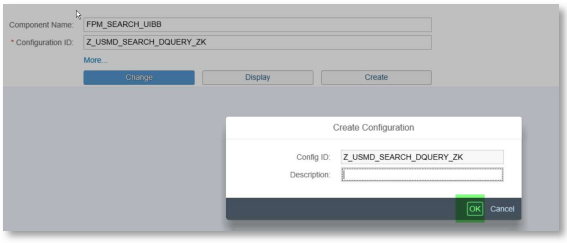
Define WF Step Processors (for Standard Workflow Template WS46000027)

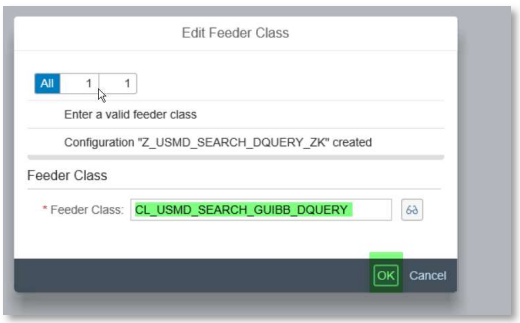
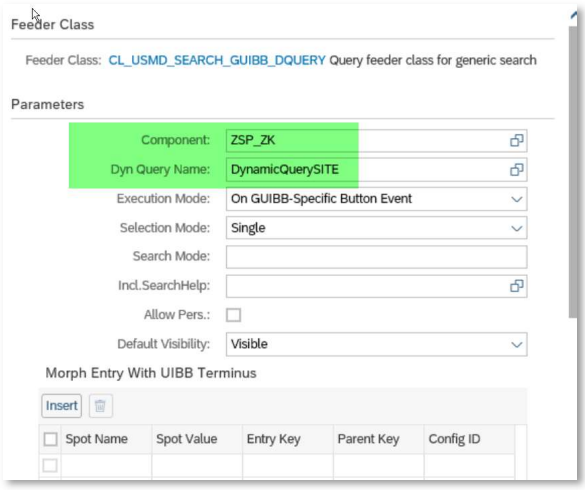
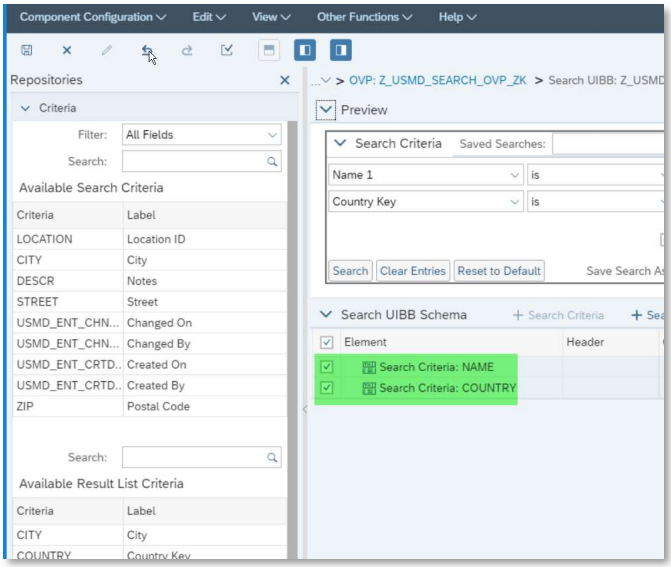
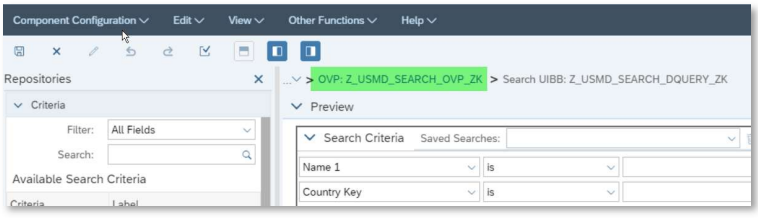
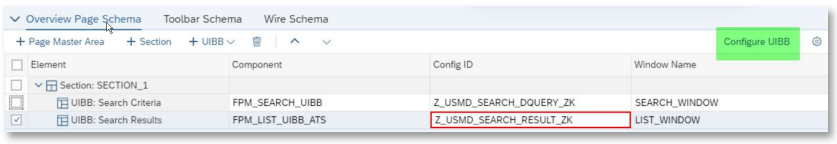
Navigate to *Assign Processor to Change Request Step Number (Simple Workflow)*	
Add steps 0-2 for your CR type (ZKCREATE), assign an *Object* US (User) and *Agent ID* (User ID) Use your own user instead of the one shown on the screen.	

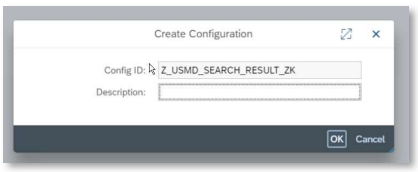
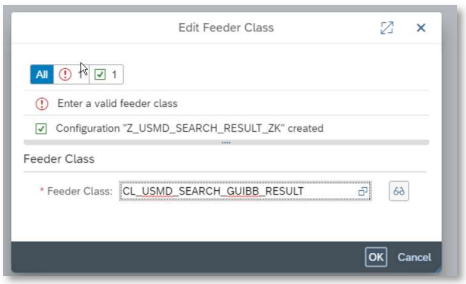
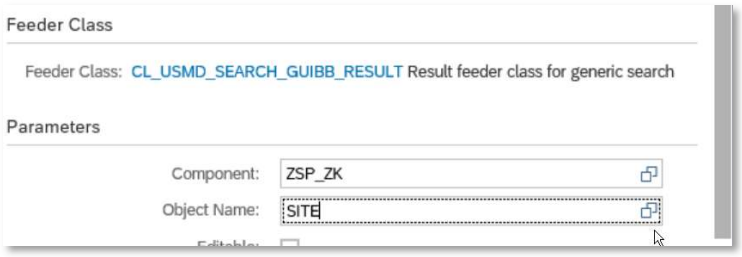
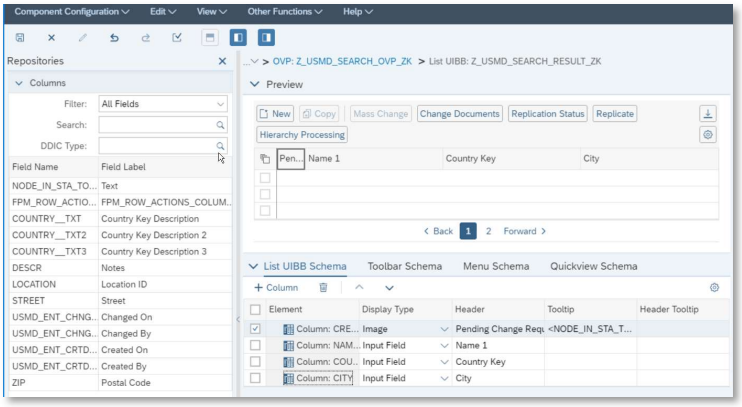
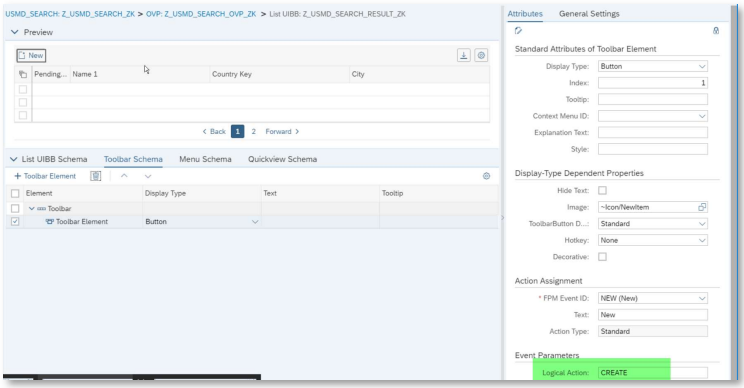
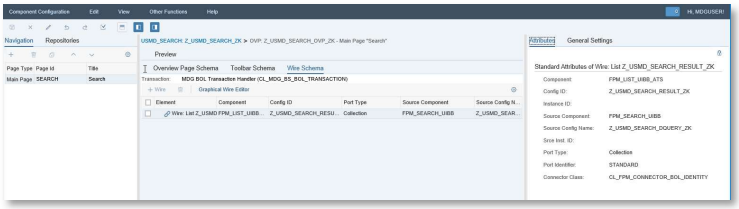
User Interface

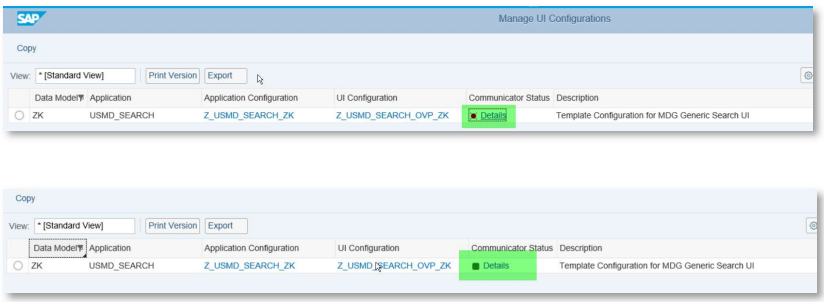
Search UI

Start MDGIMG and navigate to *Manage UI Configurations*	
Select USMD_SEARCH – USMD_SEARCH_TEMPLATE and click on *Copy*	
Deselect the rows for FPM_SEARCH_UIBB and FPM_LIST_UIBB_ATS Change *Target Configuration ID*: App_Config: Z_USMD_SEARCH_ZK UI Config: Z_USMD_SEARCH_OVP_ZK Click on *Start Deep-Copy*	 Make sure the *Change Affixes* popup doesn't come up! Place the cursor at Z_USMD_SEARCH_RESULT_ZK to make sure it works. If the *Change Affixes* popup does appear for some reason, click *Cancel* and click *Start Deep-Copy* again.
Assign a package (like $tmp)	
Navigate to the App Config	

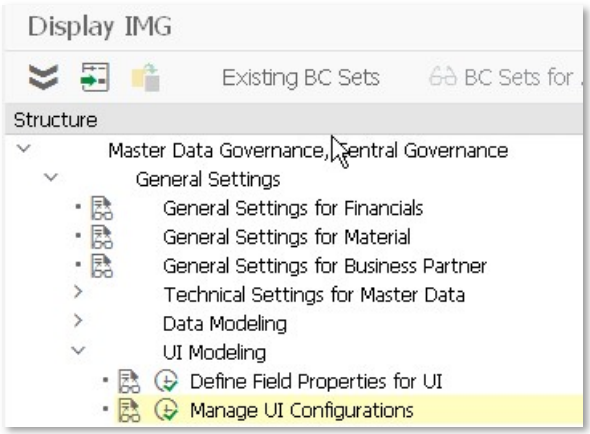
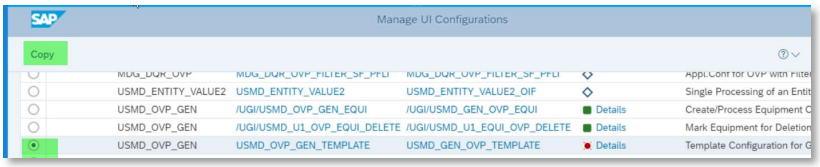
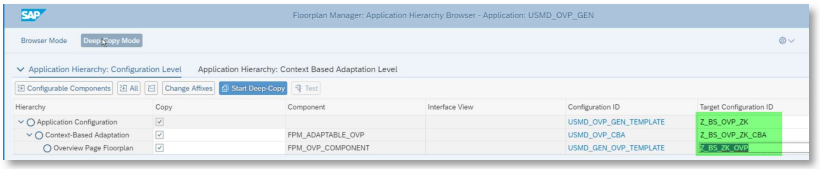
Go to *Edit* mode and change the OTC, Data Model and Entity on this level as follows: **OTC: ZK_BO1** **USMD_MODEL: ZK** **USMD_ENTITY: Site** Save your changes	
Navigate to the UI Config	
On the *Overview Page Schema* tab, rename the Config IDs for *Search Criteria* and *Search Results* using these names: **Z_USMD_SEARCH_DQUERY_ZK** **Z_USMD_SEARCH_RESULT_ZK** Save the changes	
Select the *Search Criteria* line and click on *Configure UIBB*	
A *Create Configuration* popup appears, which you confirm by choosing *OK*.	

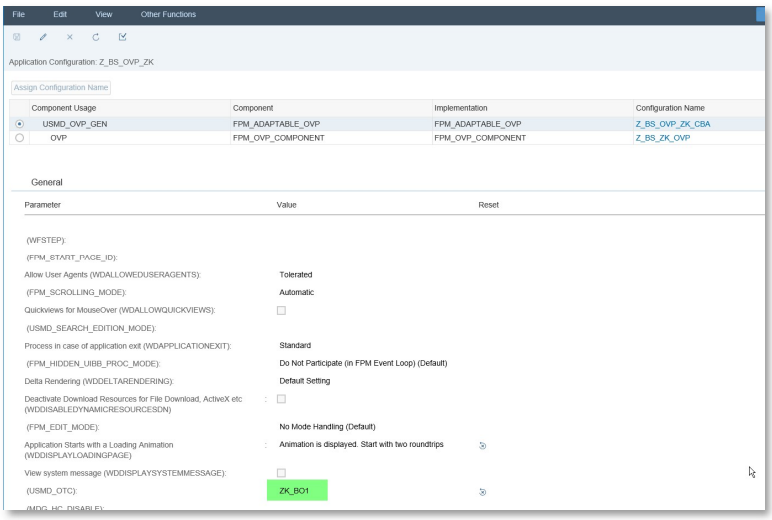
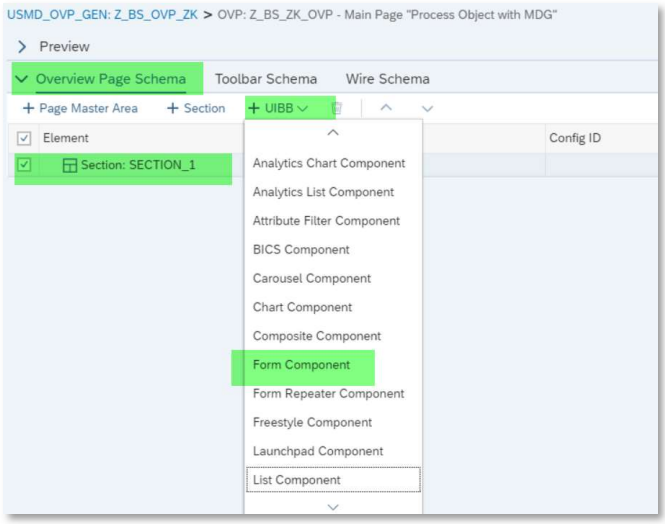
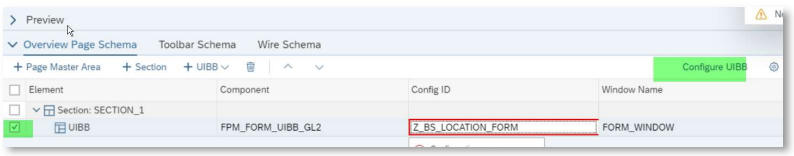
Enter the feeder class: **CL_USMD_SEARCH_GUIBB_DQUERY**	
Change the settings for the Feeder Class and Parameters by entering the following: **Component: ZSP_ZK** **Dyn Query Name: DynamicQuerySITE**	
After changing the Feeder Class Parameters, you can model your search screen. Remove Search Criteria lines as you require (see example on the right). Save your settings	
Go back to the *Search* OVP	
Select the *Search Results* line and click *Configure UIBB*	

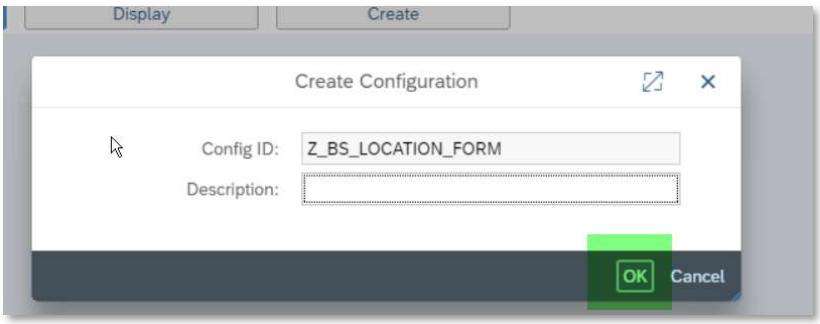
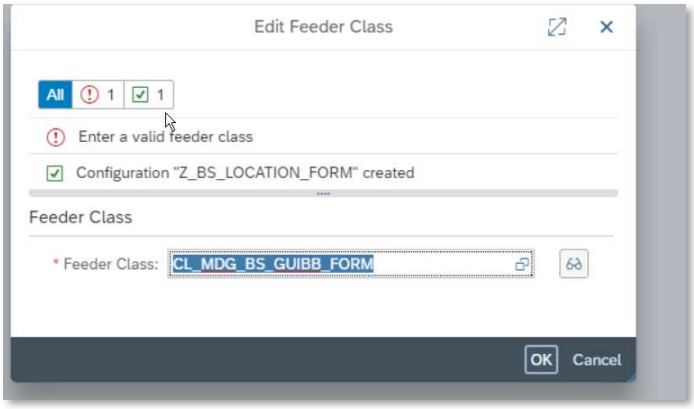
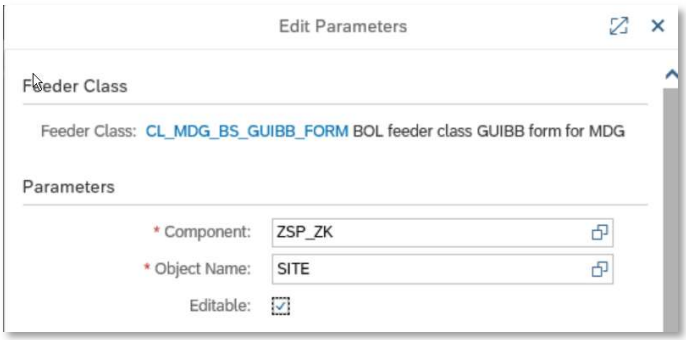
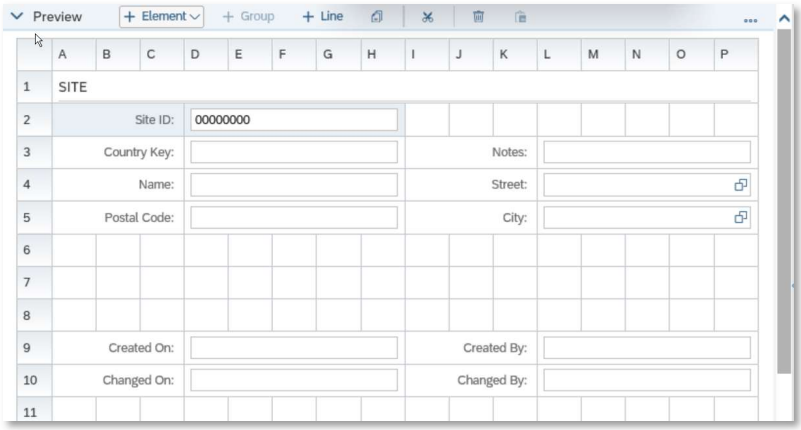
A *Create Configuration* popup appears, which you confirm by choosing *OK*.	
Define the feeder class: **CL_USMD_SEARCH_GUIBB_RESULT**	
Define Parameters as follows: **Component: ZSP_ZK** **Object Name: SITE**	
Design the Search Results columns as you require (see example on the right). Save your settings	
On the *Toolbar Schema* tab you may remove all buttons except the first one. The first one is the *New* button to trigger the creation of a new object from the search result. All other functions are not part of this guide.	
Go back to the OVP and define *Wire Schema* on OVP level as follows: **Component: FPM_LIST_UIBB_ATS** **Config ID: Z_USMD_SEARCH_RESULT_ZK** **Source Component:**	

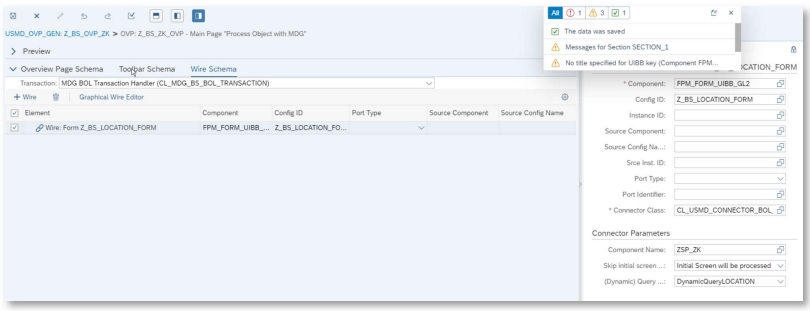
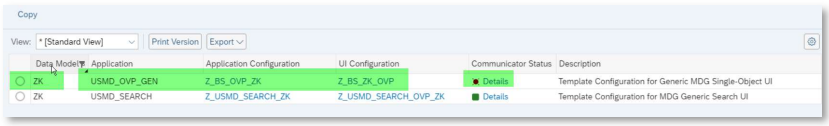
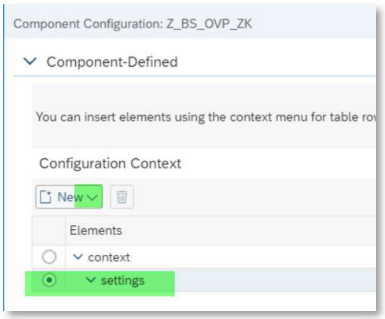
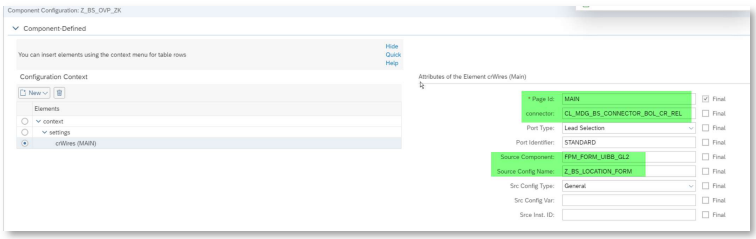
FPM_SEARCH_UIBB **Source Config Name:** **Z_USMD_SEARCH_DQUERY_ZK** Review the *Attributes* of the wire: **Port Type: Collection** **Port Identifier: STANDARD** **Connector Class:** **CL_FPM_CONNECTOR_BOL_IDENTITY**	
Create a *Communicator* for Search: Start MDGIMG again and navigate to *Manage UI Configurations.* Click the red icon of your Search configuration application. Click on the link and create an empty communicator. The icon has then changed to green.	

Single Processing UI

Start MDGIMG and navigate to *Manage UI Configurations*	
Choose *Copy*	
Use the following names: Z_BS_OVP_ZK Z_BS_OVP_ZK_CBA Z_BS_ZK_OVP Click on *Start Deep-Copy*	 Make sure the *Change Affixes* popup doesn't come up! Place the cursor at Z_USMD_SEARCH_RESULT_ZK to make sure it works. If the *Change Affixes* popup does appear for some reason, click *Cancel* and then click *Start Deep-Copy* again

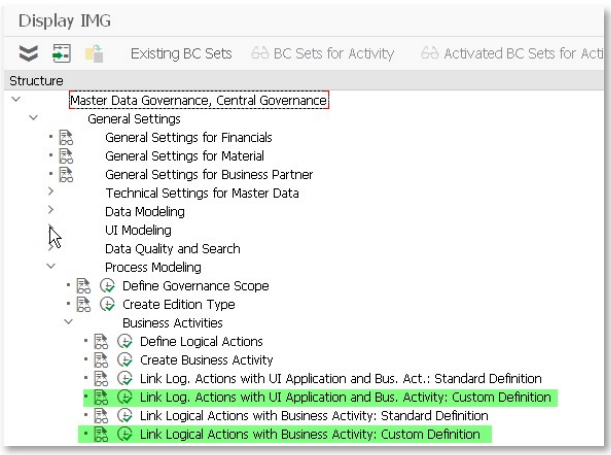
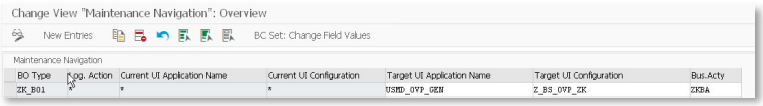
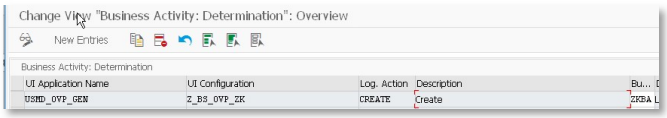
Click on the highest level Z_BS_OVP_ZK Only change the USMD_OTC entry at this level, as follows: **USMD_OTC: ZK_BO1** Save your settings	
Navigate to Z_BS_ZK_OVP Add a new *Form Component* UIBB to the section. Use the following name: Z_BS_LOCATION_FORM	  Ignore the error for now.
Select the line with the UIBB and click on *Configure UIBB*	

Confirm the popup	
Define the feeder class: **CL_MDG_BS_GUIBB_FORM**	
Define the Parameters: **Component: ZSP_ZK** **Object Name: SITE** Select the *Editable* checkbox. Confirm your entries by choosing *OK*.	
Design your screen as you require (see example shown)	 Note: You can rename the label from *Location ID* to *Site ID*
Save your changes for the Form UIBB	
Go back to the OVP	

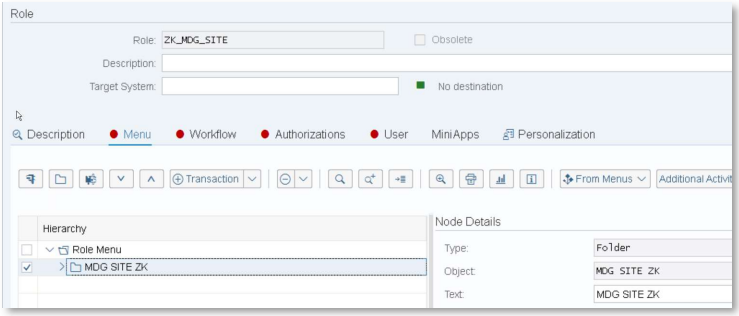
Define the wiring on the *Wire Schema* tab as follows: **Component:** **FPM_FORM_UIBB_GL2** **Config ID:** **Z_BS_LOCATION_FORM** **Connector Class:** **CL_USMD_CONNECTOR_BOL_QRY** After entering the connector you will see additional attributes: **Component Name: ZSP_ZK** **Query Name: DynamicQuerySITE**	
Save your changes	
Start *Manage UI Configurations* again and create the communicator for the new OVP. The communicator makes sure that the CR Header is added. Click on the red icon	
Create the new object and confirm.	
Select *settings* on the left-hand side and add *crWires*	 
Define the details exactly as follows: **Page ID: MAIN** **(case sensitive!)** **Connector:** **CL_MDG_BS_CONNECTOR_BOL_CR_REL** **Source Component:** **FPM_FORM_UIBB_GL2** **Source Config Name:** **Z_BS_LOCATION_FORM**	
Save your changes	

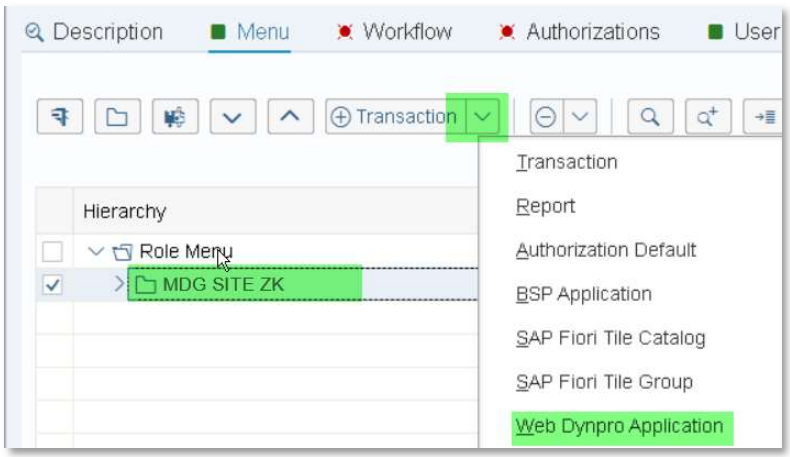
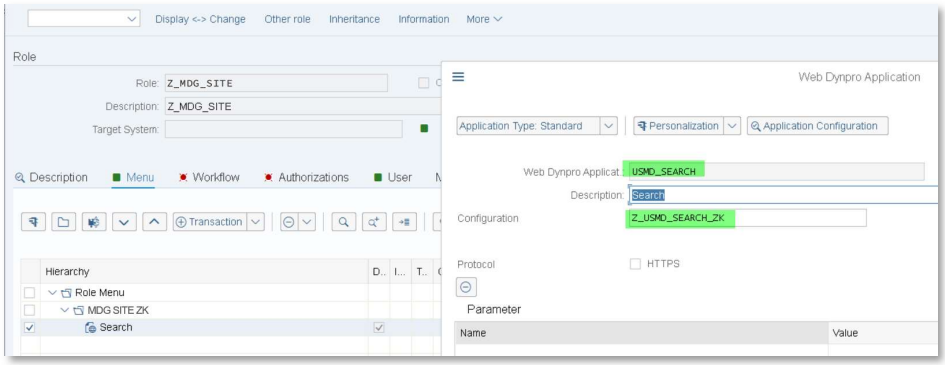
Link UIs to Actions

Add the following 2 entries (otherwise the *New* button will not work, for example!):

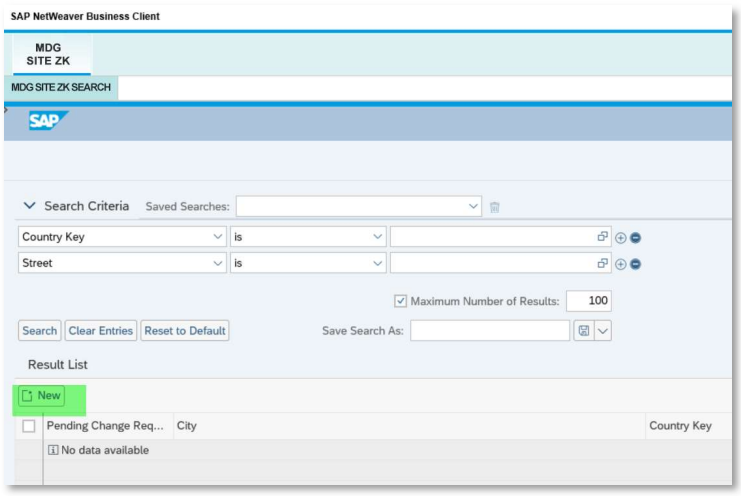
Start MDGIMG and navigate to *Link Log. Actions with UI Application and Bus. Activity: Custom Definition* Enter the following: **BO Type: ZK_BO1** **Log Action: *** **Current UI App: *** **Current UI Config: *** **Target UI App : USMD _OVP_GEN** **Target UI Config: Z_BS_OVP_ZK** **Bus. Acty: ZKBA** Save your entries	 
Start MDGIMG and navigate to *Link Logical Actions with Business Activity: Custom Definition* Enter the following: **UI App: USMD _OVP_GEN** **UI Config: Z_BS_OVP_ZK** **Log. Action: CREATE** **Bus. Acty: ZKBA** Save your entries	

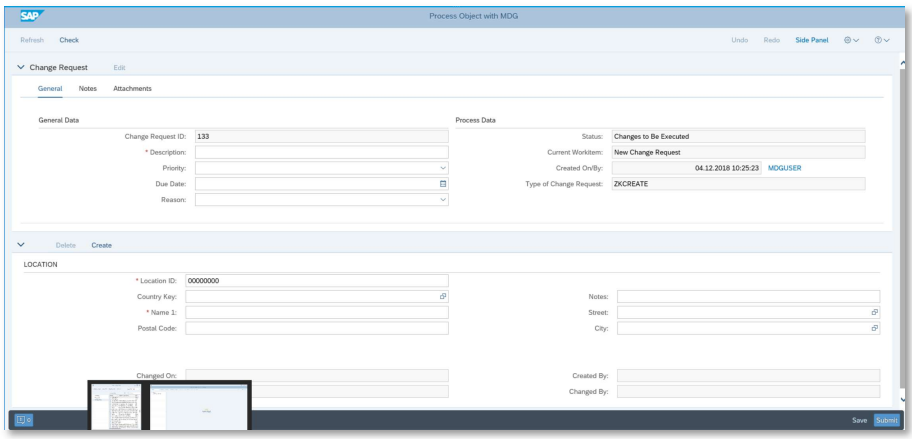
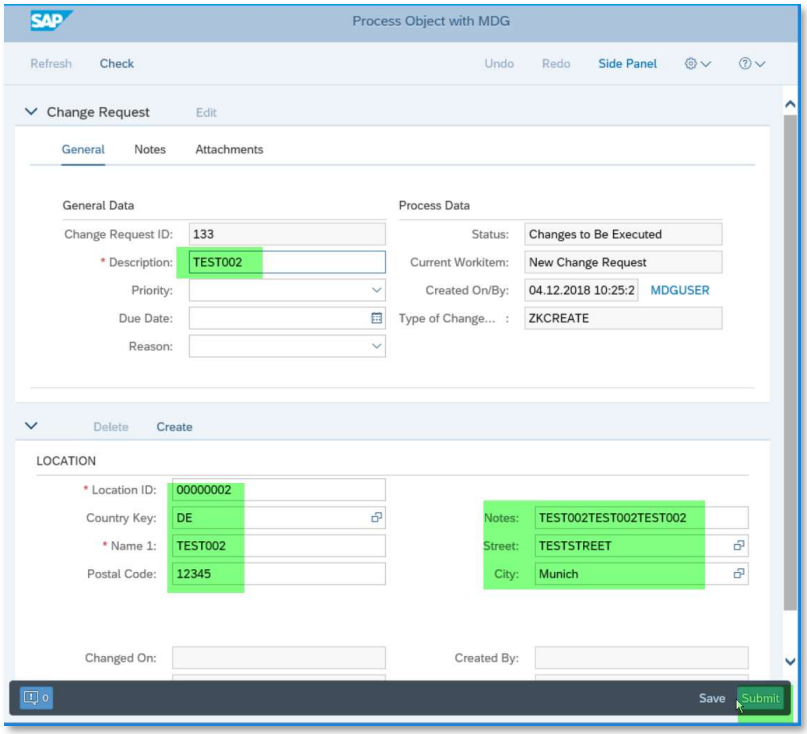
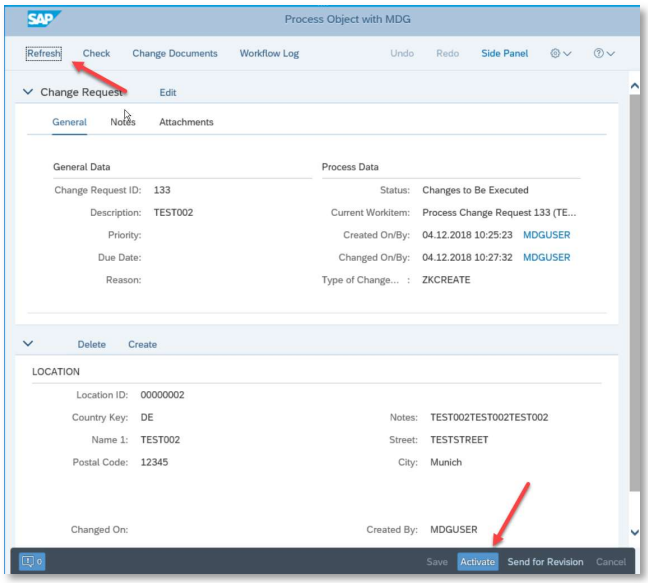
Role Model

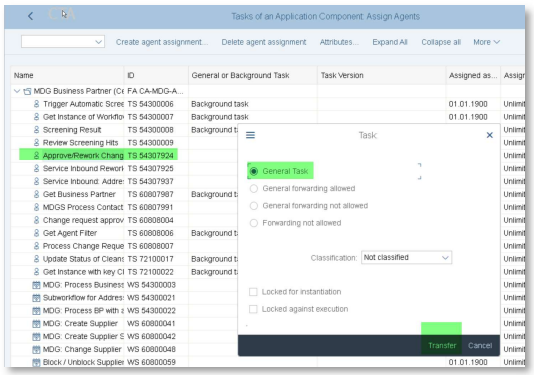
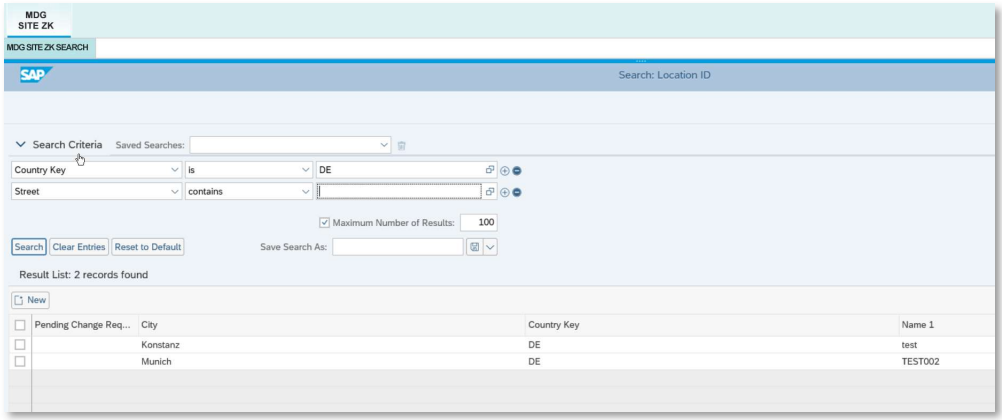
In your SAP GUI system, call transaction PFCG	
Create a new role from scratch with the name ZK_MDG_SITE	
On the *Menu* tab, create a new folder with the name: MDG SITE ZK	

Within this folder: Create a new transaction by choosing *Transaction → Web Dynpro Application*	
Enter the following details: USMD_SEARCH Z_USMD_SEARCH_ZK	
Important: Assign your test user to the role	
Save the role	

Test

Start NWBC	
Select role Z_MDG_SITE	The UI appears: 

Click on *New*	
Enter some details and Submit the change request	
Refresh & Activate	

	Info: If the task assignment (log no current processor) fails, please check if the base configuration of MDG is done. You might need to configure the task 54307924 as a general task as shown here: 
Search for the new record	

NEXT STEPS

To enrich the application, you may want to do the following things:

1. Add more entities and attributes to the data model and use relationships
2. Build a Change Request Type for log. action CHANGE to be able to modify existing records
3. Use multi-record processing mode
4. Use CBA to have a more dynamic UI
5. Introduce an auto ID for the Site/Location ID
6. Switch to a reuse model

APPENDIX

OUT OF SCOPE

This guide does not include topics like:

- Data Quality features with BRFplus or BAdIs
- Reporting capabilities

HINTS & ADDITIONAL INFORMATION

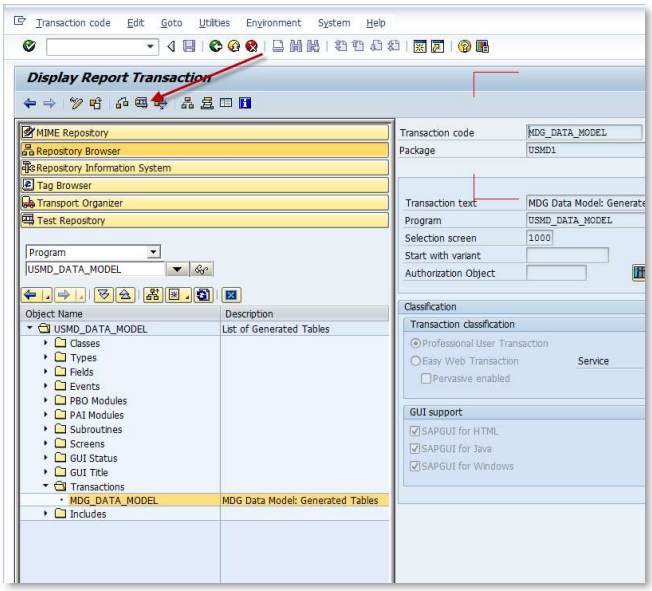
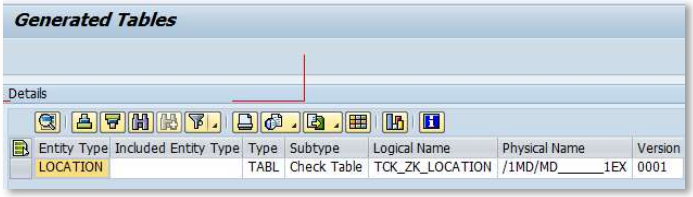
Helpful Transactions

Tcode USMD_DELETE_CREQUEST

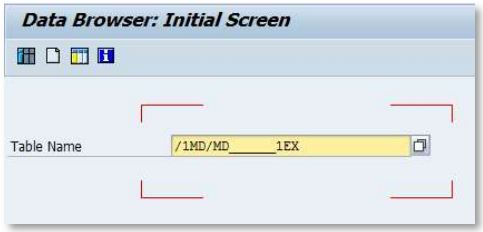
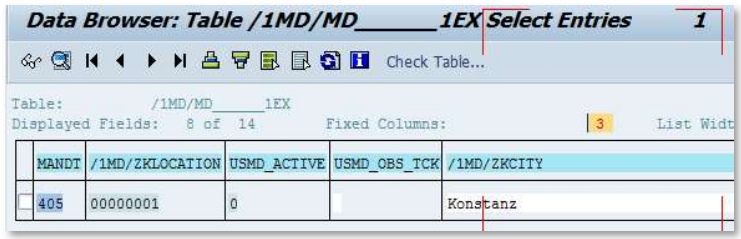
Tcode genil_model_browser

Tcode USMD_DATA_MODEL

View Generated Data Model

1.	View generated tables: Execute report USMD_DATA_MODEL in SE80 and go into the Data Model	
2.	The generated tables are displayed.	

Display Data in Staging Table

1.	Just double-click on the Physical Name and you will get forwarded to the content	
2.	Alternative method: Display table entries. Call up SE16 and enter a technical table name	
3.	Choose *Execute*	
4.	Inspect the result	

Tcode USMD_DELETE_DATA_MODEL

Delete a Data Model

If you want to delete a data model (and all dependent objects), you can do the following:

1. Make sure you are in the cross-system client.
2. Call up MDGIMG → *General Settings* → *Data Modeling* → *Edit Data Model*.
3. Select the corresponding data model and delete the line.
4. If this is not possible due to an active version, call up SE80 and run USMD_DELETE_DATA_MODEL. This will delete the active version and all dependent objects (except UI Configurations).
5. Repeat Step 3.
6. Important: You must SAVE and EXIT the MDGIMG when you're done.

Browser

Depending on which version you are using, you might experience some issues displaying the content with Internet Explorer. If so, using Google Chrome usually works.