

SAP S/4 HANA MASTER DATA GOVERNANCE

2021-03

MASTER DATA GOVERNANCE, Custom Data Model
DEMO EXPERIENCE

Creation of a Custom Data Model for a WBS

Chris Kavanagh



SAP
®

Table of Contents

Creating a Custom Data Model for a Work Breakdown Structure	3
Project System in S/4HANA	3
Create a Custom Data Model	5
<i>Concepts and Prerequisites</i>	<i>5</i>
Step by Step Procedure	7
<i>Create Business Object Type Code (OTC)</i>	<i>7</i>
Creating Entities	7
Creating Business Objects	13
Establishing the Relationships	15
Adding entities to the WBS element entity	17
Representing the Work Breakdown Structure Hierarchically	18
Creating a Custom User Interface	20
<i>User Interface Framework</i>	<i>20</i>
Develop the Search User Interface	21
Develop a Single Object Maintenance User Interface	25
Configure the technical UIBB.	26
Configure form UIBB for project definition.	30
Process Modelling	37
Create a New Business Activity	37
Assign Business Activities and Logical Actions to Business Objects	38
Create Change Request Type	39
Create a workflow	41
Creating our CR Wire	42
Creating our Role	43
Testing our Data Model	45

Creating a Custom Data Model for a Work Breakdown Structure

The SAP MDG custom Object framework is provided by SAP to help users model and build MDG applications for the master data objects specific to their business. It can additionally be used for SAP master data objects that SAP haven't provided any standard MDG applications. For example, there is no out-of-the-box data model and no MDG UI applications delivered for locations, so you could use the Custom Object framework to build an SAP MDG application for locations.

Project System in S/4HANA

Project System (PS) is a project management tool that provides users with support in all phases of their enterprise project. In SAP S/4HANA, PS provides structures that can be used to model and organise projects flexibly

PS provides two structures for mapping an enterprise project:

- Work Breakdown Structures (WBS)
- Networks

A WBS is a model of the project that shows the project deliverables in hierarchical form. WBS's are used to organise a project in the form of a hierarchy and to map the structure of a project. WBS's are made up of WBS elements that are structures in various levels to produce a hierarchy model of the project activities to be carried out.

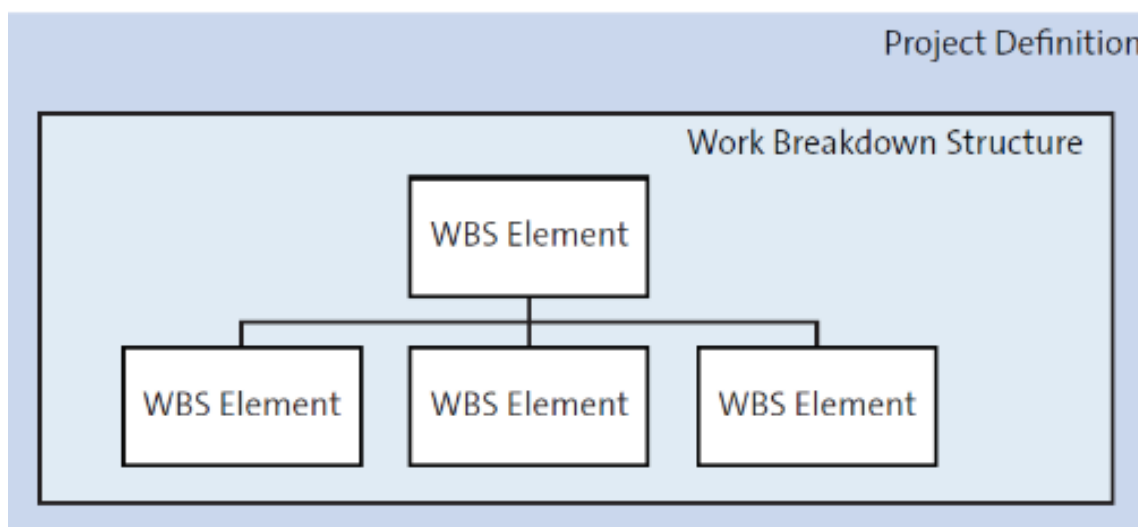
A network represents the course of a project by describing the time sequence and dependencies of events and activities in a project. Networks are used to represent project activities and logical relationships between the project activities.

The following elements are considered master data for the PS application:

- WBS elements
- Networks
- Activities.

Depending on the requirements, you can use WBSs, networks, or both to map your project in the SAP system. However, the scope of this book only includes project definition, WBSs, and WBS elements, not networks and activities

Each WBS can contain multiple WBS elements, as shown in the image below:



The following are different components of WBS

WBS elements

WBS elements represent a work package in an enterprise project. WBS elements are actual elements that are used as account assignment objects to record costs, and they can also be used as planning elements. WBS elements are arranged in a hierarchical manner, allowing the data to be summarized at any level.

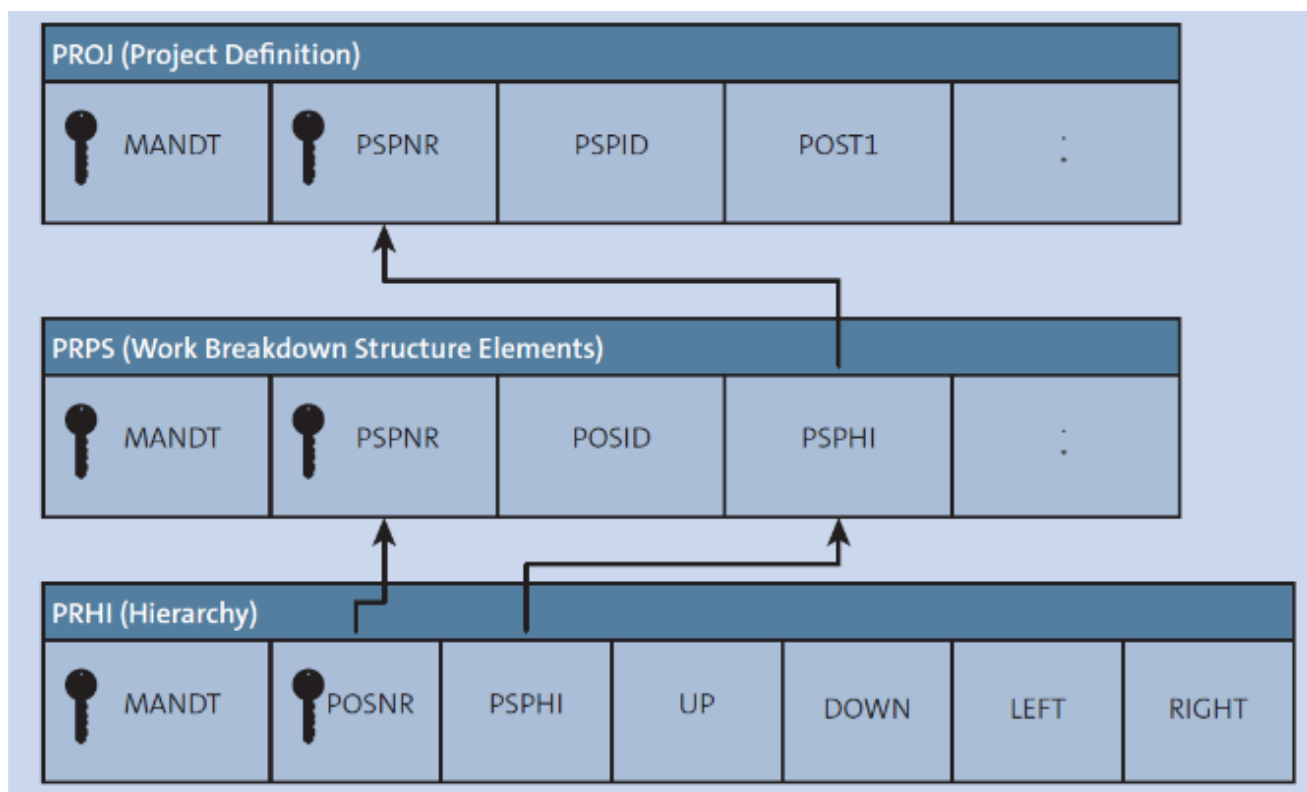
Project definition

The project definition is used to define the common attributes that are shared across the structures (WBS/network) and structure elements (WBS elements/activities) assigned to the project. Project definition is a mandatory component for creating a project with a network, WBSs, or both. The project definition holds the data that affects the whole project. For example, a controlling area entered in the project is applicable for the whole project. Project definition is also used to define organizational data such as company code, business area, profit centre, and plant. This organizational data is defaulted across the WBS elements.

WBS

The WBS is the model of the project and shows the work packages in a hierarchical structure. Each work package in an enterprise project is represented by WBS elements

Below are the various tables that together form the data model for PS. All data is saved in the tables. This database architecture will be the basis for the SAP Master Data Governance data model.



The MDG foundation framework uses the SAP MDG data model entities, attributes of entities, and relationships between entities to generate the staging area. The staging area is an exclusive persistence layer for MDG, generated from an active SAP MDG data model. The staging area is used to store both active data and inactive data. The goal of the data model is to generate these staging area tables correctly and be the single source of information for relationships between various SAP Master Data Governance entities. The MDG data model is also a source of metadata required for UI modelling. There are two storage modes for active data:

Reuse mode

This mode is used if the tables needed already exist in SAP S/4HANA. Usually these are the master data objects that are available as part of the SAP S/4HANA data model but aren't delivered as out-of-the-box SAP MDG data models. To use the reuse active area for the custom data models, you must create an active area and assign the access class to the active area.

Flex mode

This mode is used if no tables are available in SAP S/4HANA. Usually these are the master data objects that aren't available as part of the standard SAP S/4HANA data model. Ideally, the flex option is preferred for business objects that require edition management. SAP S/4HANA doesn't have the edition concept; instead, these layers enforce time dependency by using valid-from date, valid-to date, or both as key fields in the table.

In our example of WBSs in PS, we'll choose flex mode because the flex model doesn't require the creation of an active area access class, and our current data model involves a hierarchy.

Create a Custom Data Model

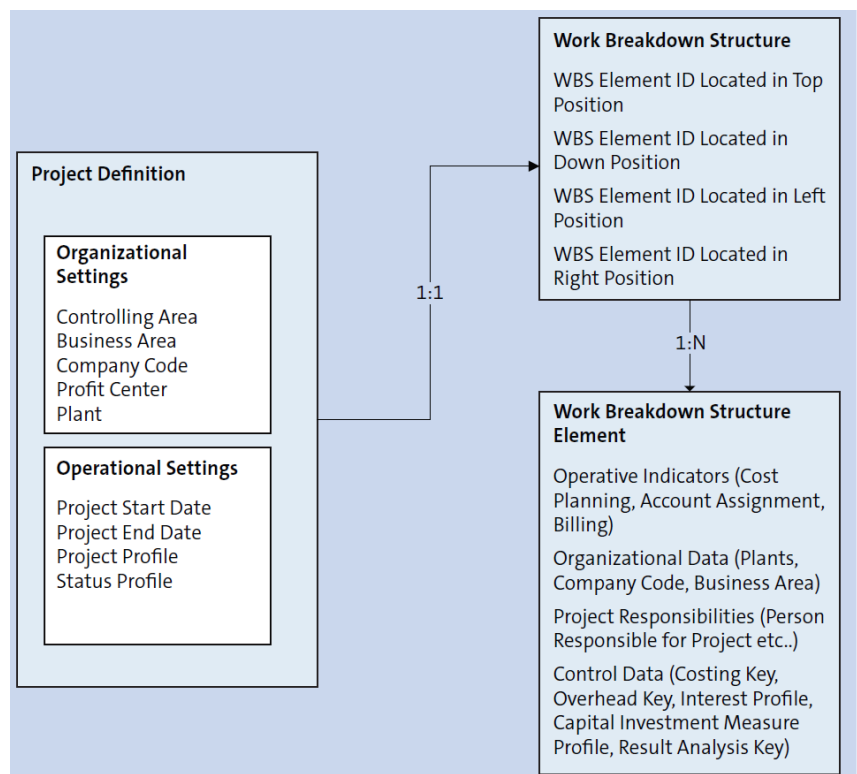
The first step toward building custom SAP MDG applications is to create a custom data model. The Generic Interaction Layer (GenIL) provides uniform API services to access and manipulate underlying business data. The Business Object Layer (BOL) consumes the GenIL API. The following sections explain how to create the custom data model, entities, and relationships in detail.

Concepts and Prerequisites

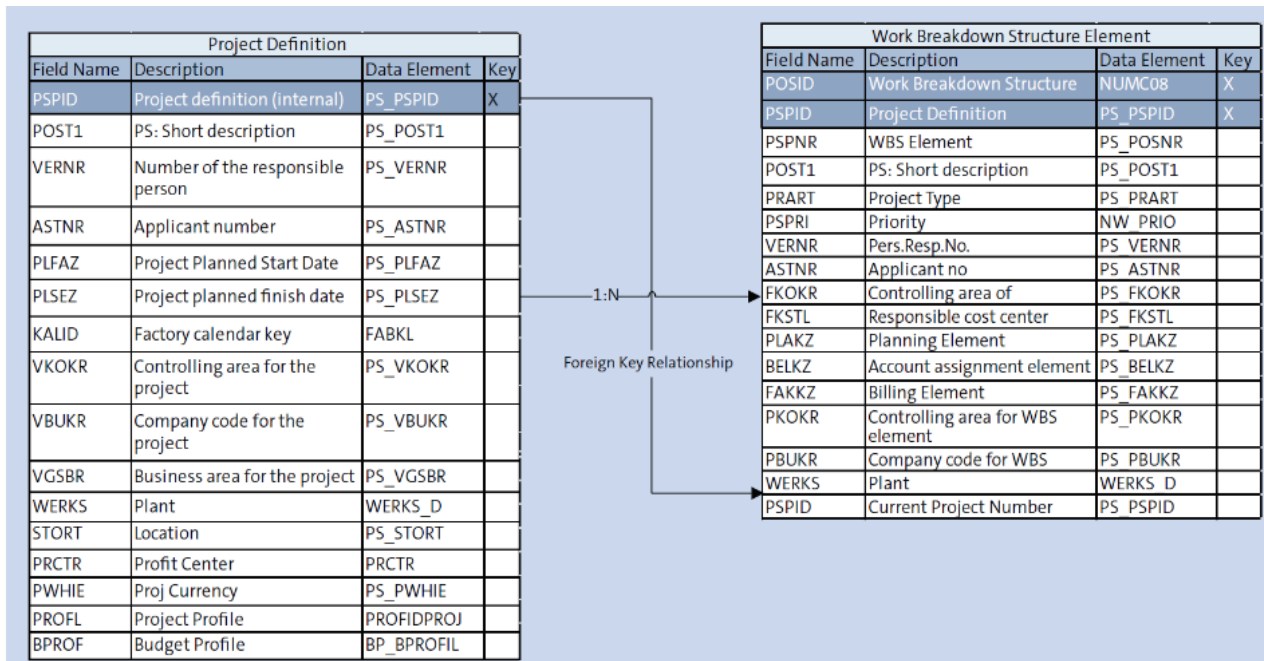
The process of creating entities and attributes for our custom data model ZX will be described via the following:

- Conceptual data model
- Logical data model
- Physical data model

First the conceptual data model: This data model is a high-level representation of the data model architecture. The conceptual model is created with non-technical names make it easy to understand and process for project stakeholders such as executives, business users, and business subject matter experts (SME's). The conceptual data model acts as the basis for creating the logical data model.



The logical data model is a more technical representation of the data model, it displays entities attributes and relationships. The logical data model is normalized by specifying the field-level details such as data type and data length. Below is the logical data model for our custom data model; you can see project definition and WBS elements along with detailed attributes. The target audience for the logical data model, is business SME's, expert modelers, and application experts.



Finally, the physical data modelling is the process of creating the actual data model in the system. During this step, you take the outcomes (e.g., design document) from the preceding two steps and implement them in the system. The physical data model is system specific and deeply technical in nature.

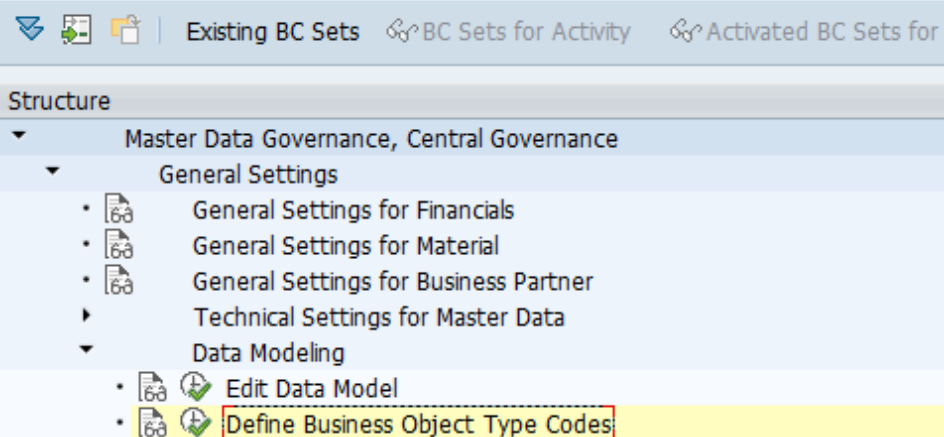
Step by Step Procedure

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.

Display IMG



Select New Entries in the top right corner to create your BO Type.

Following the correct naming conventions such as Z*.

EG: ZCKPS

Fill in description.

Business Object Type Code	
BO Type	Description
ZCKPS	Project Structure CK

Save Your changes

Creating Entities

In the following sections, we'll look at creating entities for your custom data model, starting with type 1 entities before moving on to type 3.

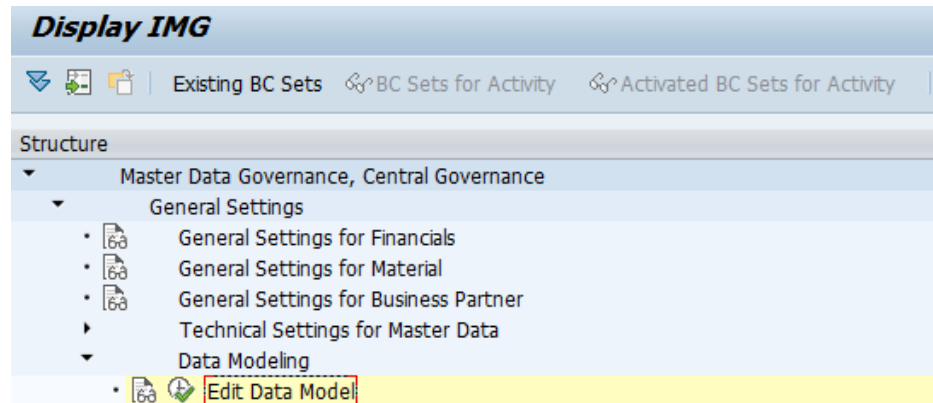
Every data model should have at least one type 1 entity. In the current example, the project definition is one of the type 1 entities.

Return to the MDGIMG screen.

Navigate to 'Edit Data Model'.

General Settings →
Data Modeling →
Edit Data Model.

*Alternatively, you can use
the Configuration workbench*



Click New entries to create a new Data Model.

Fill in the details as follows

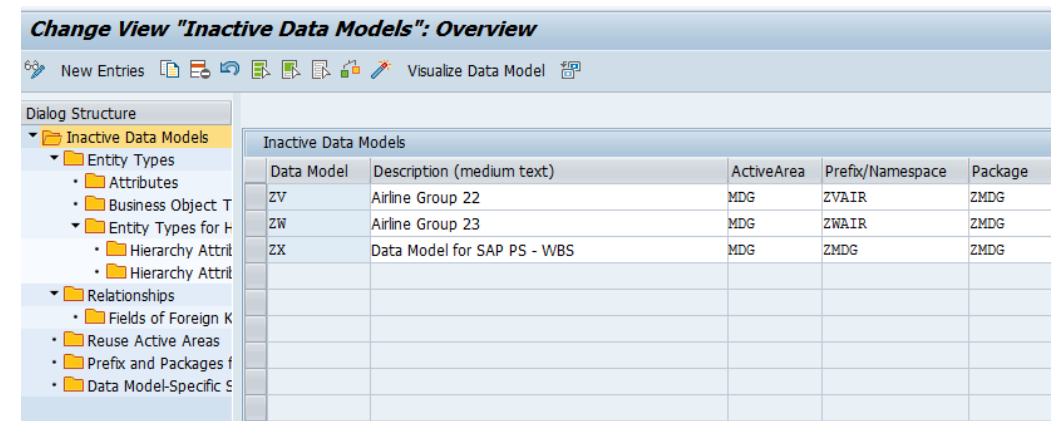
Data Model: ZX

Descr.: Data Model for SAP PS - WBS

Active Area: provided by MDG

Namespace: ZMDG

Package: ZMDG



Select new entries and add your first entity PSPID.

Storage: Type 1

Data Element: PS_PSPID

1. Edition management as been disabled. In the underlying SAP S/4HANA data model, PS isn't a time dependent entity
2. Every type 1 entity needs a key assigned.
3. The project definition can have language dependent texts.
4. The Active area field has been left blank; when left blank the active area of the data model is adopted

Click on Attributes tab and add New attributes for the domain:

See the picture to the right and fill in the attributes as shown.

Data Model	ZX		
Entity Type	PSPID		
Attributes			
Attribute	Key Field	Data Element	Description
ASTNR	<input type="checkbox"/>	PS_ASTNR	Application Number
KALID	<input type="checkbox"/>	FABKL	Factory Calendar key
PLFAZ	<input type="checkbox"/>	PS_PLFAZ	Projected Planned Start Date
PLSEZ	<input type="checkbox"/>	PS_PLSEZ	Projected planned finish Date
POST1	<input type="checkbox"/>	PS_POST1	PS: Short Description
PROFL	<input type="checkbox"/>	PROFIDPROJ	Project Profile
PWHIE	<input type="checkbox"/>	PS_PWHIE	Proj Currency
VERNR	<input type="checkbox"/>	PS_VERNR	Number Of the Responsible Person
VGSBR	<input type="checkbox"/>	PS_VGSBR	Business Area for the project

Save your changes

Select new Entries as Previous

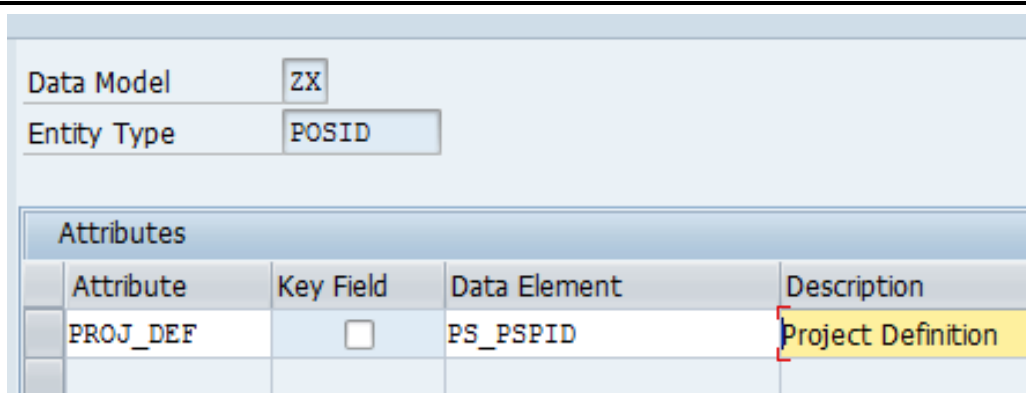
We'll be creating a WBS element as another type 1 entity, but we will assign it the same business object as the project definition (PSPID). By assigning it to the same business object as the project definition, we're ensuring that a WBS element can't be replicated alone, and it needs to be replicated through the project definition.

Fill in the Entity details as shown in the picture

Data Model	ZX
Entity Type	PSPID
Entity Types	
Storage/Use Type	Changeable via Change Request; Generated Database Tables
Validity / Entity	No Edition
Data Element	PS_POSID
Is Hry Type	No
Validity / Hierarchy	No Edition
Key Assignment	Key Cannot Be Changed; No Internal Key Assignment
<input checked="" type="checkbox"/> Language-Dep. Texts	
Long Text: Length	60
Medium Text: Length	
Short Text: Length	
<input type="checkbox"/> Attachments	
<input type="checkbox"/> Sets	
Search Help	
Src. Fld Short Text	
Src. Fld Medium Text	
Src. Fld Long Text	
Temporary Keys	
Active Area	
Deletion	Deletion Allowed
Description	WBS Elements

Select attributes

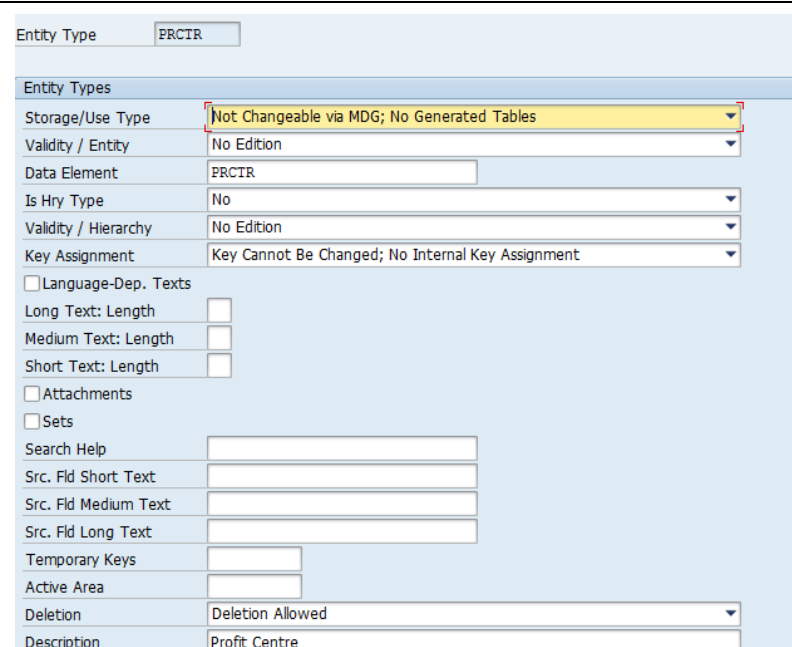
We will be adding an attribute PROJ_DEF, this attribute is used to store the project definition ID of the project to which the WBS element is assigned. This attribute needs to have a foreign key relationship with PSPID.



Data Model			
Entity Type		POSID	
Attributes			
Attribute	Key Field	Data Element	Description
PROJ_DEF	<input type="checkbox"/>	PS_PSPID	Project Definition

Save the changes

The project definition (PSPID) has profit centre as one of the attributes. We cant add profit centre as an attribute directly under the type 1 Entity PSPID because the SAP MDG data model rules dictate that a check table of an entity's attribute cant have more key fields besides the client and key field referring to the attribute.



Entity Types	
Storage/Use Type	Not Changeable via MDG; No Generated Tables
Validity / Entity	No Edition
Data Element	PRCIR
Is Hry Type	No
Validity / Hierarchy	No Edition
Key Assignment	Key Cannot Be Changed; No Internal Key Assignment
<input type="checkbox"/> Language-Dep. Texts Long Text: Length <input type="checkbox"/> Medium Text: Length <input type="checkbox"/> Short Text: Length <input type="checkbox"/> <input type="checkbox"/> Attachments <input type="checkbox"/> Sets Search Help <input type="text"/> Src. Fld Short Text <input type="text"/> Src. Fld Medium Text <input type="text"/> Src. Fld Long Text <input type="text"/> Temporary Keys <input type="text"/> Active Area <input type="text"/> Deletion <input type="text"/> Deletion Allowed Description <input type="text"/> Profit Centre	

The controlling area is present as an attribute both in the project definition and in the WBS element. Therefore, it makes sense to create just one type 3 entity and then assign the same entity as an attribute to the project structure and WBS element entities using referencing relationships.

The controlling area is one of the key fields in the check tables assigned to the profit centre therefore, it's imperative that we create the controlling area as a type 2 entity so that the entity can be used to establish a leading relationship.

Entity Type **Con Area**

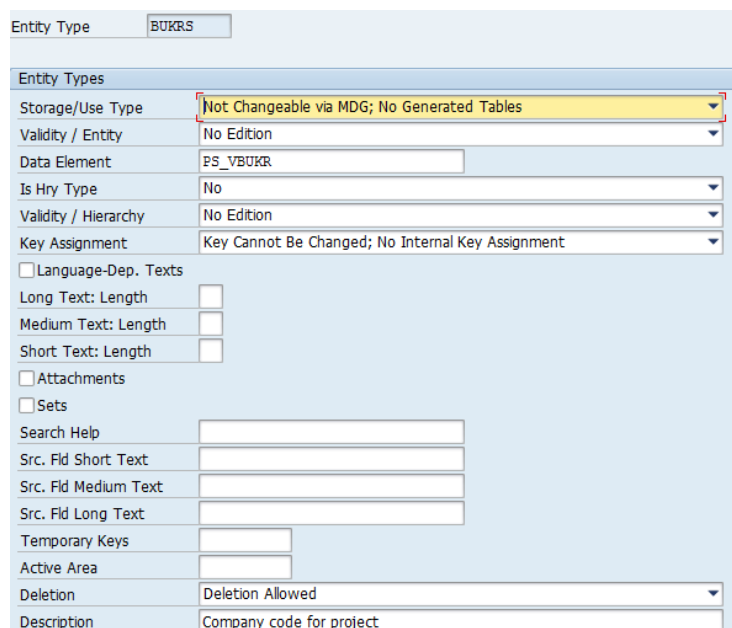
Entity Types	
Storage/Use Type	Not Changeable via MDG; No Generated Tables
Validity / Entity	No Edition
Data Element	PS_VKOKR
Is Hry Type	No
Validity / Hierarchy	No Edition
Key Assignment	Key Cannot Be Changed; No Internal Key Assignment
<input type="checkbox"/> Language-Dep. Texts	
Long Text: Length	<input type="checkbox"/>
Medium Text: Length	<input type="checkbox"/>
Short Text: Length	<input type="checkbox"/>
<input type="checkbox"/> Attachments	
<input type="checkbox"/> Sets	
Search Help	<input type="text"/>
Src. Fld Short Text	<input type="text"/>
Src. Fld Medium Text	<input type="text"/>
Src. Fld Long Text	<input type="text"/>
Temporary Keys	<input type="text"/>
Active Area	<input type="text"/>
Deletion	Deletion Allowed
Description	Controlling Area

Like the controlling area the Valid to date is a key attribute in the check table so we will create it as entity 3 so we can establish a leading relationship.

Entity Type **DATBI**

Entity Types	
Storage/Use Type	Not Changeable via MDG; No Generated Tables
Validity / Entity	No Edition
Data Element	DATBI
Is Hry Type	No
Validity / Hierarchy	No Edition
Key Assignment	Key Cannot Be Changed; No Internal Key Assignment
<input type="checkbox"/> Language-Dep. Texts	
Long Text: Length	<input type="checkbox"/>
Medium Text: Length	<input type="checkbox"/>
Short Text: Length	<input type="checkbox"/>
<input type="checkbox"/> Attachments	
<input type="checkbox"/> Sets	
Search Help	<input type="text"/>
Src. Fld Short Text	<input type="text"/>
Src. Fld Medium Text	<input type="text"/>
Src. Fld Long Text	<input type="text"/>
Temporary Keys	<input type="text"/>
Active Area	<input type="text"/>
Deletion	Deletion Allowed
Description	Valid to Date

Next, we will create the company code as a type 3 entity as an attribute to PSPID using a referencing relationship. We are choosing Type 3 as we later want to establish it a leading relationship with the plant entity.



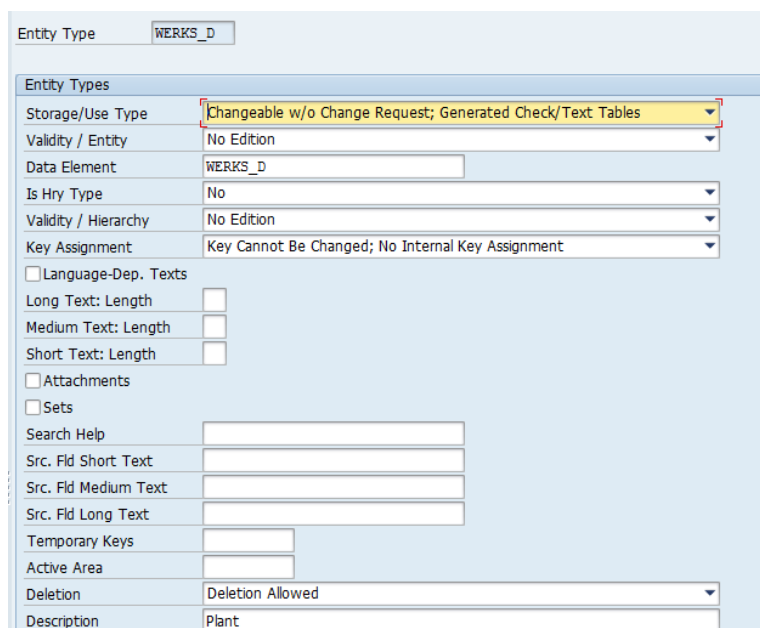
Entity Type: BUKRS

Entity Types

Storage/Use Type	Not Changeable via MDG; No Generated Tables
Validity / Entity	No Edition
Data Element	PS_VBUKR
Is Hry Type	No
Validity / Hierarchy	No Edition
Key Assignment	Key Cannot Be Changed; No Internal Key Assignment
<input type="checkbox"/> Language-Dep. Texts	
Long Text: Length	<input type="checkbox"/>
Medium Text: Length	<input type="checkbox"/>
Short Text: Length	<input type="checkbox"/>
<input type="checkbox"/> Attachments	
<input type="checkbox"/> Sets	
Search Help	
Src. Fld Short Text	
Src. Fld Medium Text	
Src. Fld Long Text	
Temporary Keys	
Active Area	
Deletion	Deletion Allowed
Description	Company code for project

Plant is used to define organization data during project definition.

To ensure that the SAP Master Data Governance framework understands this relationship between plant and location, we'll create plant as a type 3 entity because we need it to be part of the leading relationship for the location attribute.



Entity Type: WERKS_D

Entity Types

Storage/Use Type	Changeable w/o Change Request; Generated Check/Text Tables
Validity / Entity	No Edition
Data Element	WERKS_D
Is Hry Type	No
Validity / Hierarchy	No Edition
Key Assignment	Key Cannot Be Changed; No Internal Key Assignment
<input type="checkbox"/> Language-Dep. Texts	
Long Text: Length	<input type="checkbox"/>
Medium Text: Length	<input type="checkbox"/>
Short Text: Length	<input type="checkbox"/>
<input type="checkbox"/> Attachments	
<input type="checkbox"/> Sets	
Search Help	
Src. Fld Short Text	
Src. Fld Medium Text	
Src. Fld Long Text	
Temporary Keys	
Active Area	
Deletion	Deletion Allowed
Description	Plant

The check table assigned to location has plant and location as key fields. Therefore, as explained before, to be compliant with the SAP MDG data model rules we must:

- Create a location as type 3 entity
- Create a leading relationship between the entity for plant and the entity for location.
- Create a referencing relationship between the entity for location and entity PSPID

Data Model

Entity Type

Entity Types

Storage/Use Type	Changeable w/o Change Request; Generated Check/Text Tables
Validity / Entity	No Edition
Data Element	PS_STORT
Is Hry Type	No
Validity / Hierarchy	No Edition
Key Assignment	Key Cannot Be Changed; No Internal Key Assignment
<input type="checkbox"/> Language-Dep. Texts	
Long Text: Length	<input type="text"/>
Medium Text: Length	<input type="text"/>
Short Text: Length	<input type="text"/>
<input type="checkbox"/> Attachments	
<input type="checkbox"/> Sets	
Search Help	<input type="text"/>
Src. Fld Short Text	<input type="text"/>
Src. Fld Medium Text	<input type="text"/>
Src. Fld Long Text	<input type="text"/>
Temporary Keys	<input type="text"/>
Active Area	<input type="text"/>
Deletion	Deletion Allowed
Description	Location
Structure/Table	<input type="text"/>
Field	<input type="text"/>
Struct. X-Flds	<input type="text"/>

Save your changes and activate

Creating Business Objects

Returning to our list of entities select the first entity we created PSPID.

Open Business Object Type view from the left column.

Data Model

Entity Types			
Entity Type	Storage/Use Type	Val.Entity	Data Element
BUKRS	Not Changeable via MDG; No ...	No Edition	PS_VBUKR
CON_AREA	Not Changeable via MDG; No ...	No Edition	PS_VKOKR
DATBI	Not Changeable via MDG; No ...	No Edition	DATBI
POSID	Changeable via Change Reque...	No Edition	PS_POSID
PRCTR	Not Changeable via MDG; No ...	No Edition	PRCTR
PSPID	Changeable via Change Reque...	No Edition	PS_PSPID
STORT	Changeable w/o Change Reque...	No Edition	PS_STORT
WERKS_D	Changeable w/o Change Reque...	No Edition	WERKS_D

We can input our business object we previously created.

Important to note we have selected the root check box for this entity.

Change View "Business Object Type": Overview

New Entries | Visualize Data Model

Dialog Structure

- Inactive Data Models
 - Entity Types
 - Attributes
 - Business Object T
 - Entity Types for H
 - Hierarchy Attrib
 - Hierarchy Attrib

Data Model: ZX
Entity Type: PSPID

Business Object Type		
BO Type	Root	Description
ZCKPS	<input checked="" type="checkbox"/>	Project Structure CK

Repeating the steps as previous, return to our list of entities, this select the 2nd entity we created POSID. Then select business object Type from the left column

Change View "Entity Types": Overview

New Entries | Visualize Data Model

Dialog Structure

- Inactive Data Models
 - Entity Types
 - Attributes
 - Business Object T
 - Entity Types for H
 - Hierarchy Attrib
 - Hierarchy Attrib
 - Relationships
 - Fields of Foreign K
 - Reuse Active Areas
 - Prefix and Packages f
 - Data Model-Specific S

Data Model: ZX

Entity Type	Storage/Use Type	Val.Entity	Data Element
BUKRS	Not Changeable via MDG; No ...	No Edition	PS_VBUKR
CON_AREA	Not Changeable via MDG; No ...	No Edition	PS_VKOKR
DATBI	Not Changeable via MDG; No ...	No Edition	DATBI
POSID	Changeable via Change Reque...	No Edition	PS_POSID
PRCTR	Not Changeable via MDG; No ...	No Edition	PRCTR
PSPID	Changeable via Change Reque...	No Edition	PS_PSPID
STORT	Changeable w/o Change Reque...	No Edition	PS_STORT
WERKS_D	Changeable w/o Change Reque...	No Edition	WERKS_D

Like before input the business object we previously created. In this case leave the root box unchecked as shown

Dialog Structure

- Inactive Data Models
 - Entity Types
 - Attributes
 - Business Object T
 - Entity Types for H
 - Hierarchy Attrib
 - Hierarchy Attrib

Data Model: ZX
Entity Type: POSID









Business Object Type		
BO Type	Root	Description
ZCKPS	<input type="checkbox"/>	Project Structure CK

Save and Activate changes

Establishing the Relationships

Select Relationships from the left column. Then select new entries from the tool bar to start creating a relationship.

Change View "Relationships": Overview

New Entries        Visualize Data Model 

Dialog Structure

- Inactive Data Models
 - Entity Types
 - Attributes
 - Business Object T
 - Entity Types for H
 - Hierarchy Attrit
 - Hierarchy Attrit
 - Relationships
 - Fields of Foreign K

Data Model **ZX**

From-EntityType	Relationship	To-Entity Type	Relation. Type

As discussed, before we need a foreign key relationship through attribute PROJ_DEF between PSPID and POSID. Firstly, we will establish the relationship as shown.









Data Model **ZX**

From-EntityType	Relationship	To-Entity Type	Relation. Type	Cardinality
PSPID	ZFOR_KEY	POSID	F	1 : N

Then select Fields of Foreign keys.

Here is where we will indicate PROJ_DEF is our foreign key. Input as shown

Change View "Fields of Foreign Key Relationships": Overview

New Entries        Visualize Data Model 

Dialog Structure

- Inactive Data Models
 - Entity Types
 - Attributes
 - Business Object T
 - Entity Types for H
 - Hierarchy Attrit
 - Hierarchy Attrit
 - Relationships
 - Fields of Foreign K



















Data Model **ZX**

From-EntityType **PSPID**

Relationship **ZFOR_KEY**

"From" Field (Check Table Fld)	"To" Field (Foreign Key Field)
PSPID	PROJ_DEF

Once opened your data model should appear as shown. You can see the difference relationships that we established.

Detail View  Active Version						
Data Model		Name	Data Element	Referenced Element
▼ ZX						
▼ PSPID		Project Definition				
•  PSPID		Project definition	E...	<input checked="" type="checkbox"/>	PS_PSPID	
•  ASTNR		Applicant no.	A...	<input type="checkbox"/>	PS_ASTNR	
•  DATBI_R		Valid To	A...	<input type="checkbox"/>	DATBI	DATBI
•  KALID		Factory Calendar	A...	<input type="checkbox"/>	FABKL	
•  PLFAZ		Start date	A...	<input type="checkbox"/>	PS_PLFAZ	
•  PLSEZ		Finish date	A...	<input type="checkbox"/>	PS_PLSEZ	
•  POST1		Description	A...	<input type="checkbox"/>	PS_POST1	
•  PRCTR		Profit Center	A...	<input type="checkbox"/>	PRCTR	PRCTR
•  PROFL		Project Profile	A...	<input type="checkbox"/>	PROFIDPROJ	
•  PWHIE		Project Currency	A...	<input type="checkbox"/>	PS_PWHIE	
•  STORT		Location	A...	<input type="checkbox"/>	PS_STORT	STORT
•  TXTLG		Description (lo...	A...	<input type="checkbox"/>	USMD_TXTLG	
•  VBUKR		Company code	A...	<input type="checkbox"/>	PS_VBUKR	BUKRS
•  VERNR		No. of Person ...	A...	<input type="checkbox"/>	PS_VERNR	
•  VGSBR		Business area	A...	<input type="checkbox"/>	PS_VGSBR	
•  VKOKR		Controlling area	A...	<input type="checkbox"/>	PS_VKOKR	CON_AREA
•  WERKS		Plant	A...	<input type="checkbox"/>	WERKS_D	WERKS_D

Adding entities to the WBS element entity

We need to add attributes to our POSID entity.

Returning to our entity list select POSID and click attributes from the left column as done previously.

The select new Entries.

Change View "Attributes": Overview

New Entries

Dialog Structure

Inactive Data Models

Entity Types

Attributes

Business Object Type

Entity Types for Hierarc

Hierarchy Attributes

Hierarchy Attributes

Data Model

ZX

Entity Type

POSID

Attributes

Attribute	Key Field	Data Element
PROJ_DEF	<input type="checkbox"/>	PS_PSPID

Among the attributes that need to be added to the POSID entity there are simple attributes and modelled attributes. Simple attributes can be added through the attribute tab whereas modelled need to be added through relationships.

Shown is the list of simple attributes, add these as shown.

Data Model	ZX
Entity Type	POSID

Attributes				
	Attribute	Key Field	Data Element	Description
	ASTNR	<input type="checkbox"/>	PS_ASTNR	Applicant no
	BELKZ	<input type="checkbox"/>	PS_BELKZ	Account Assignment element
	FAKKZ	<input type="checkbox"/>	PS_FAKKZ	Billing Element
	PLAKZ	<input type="checkbox"/>	PS_PLAKZ	Planning Element
	POST1	<input type="checkbox"/>	PS_POST1	PS:Short Description
	PRART	<input type="checkbox"/>	PS_PRART	Project Type
	PSPRI	<input type="checkbox"/>	NW_PRIO	priority
	PS_POSID	<input type="checkbox"/>	PS_POSID	Pers.Resp No
	VERNR	<input type="checkbox"/>	PS_VERNR	

As before to establish relationships, select relationship from the left column and click new entries.

Add relationships as shown.

Data Model ZX					
Relationships					
	From-Entity Type	Relationship	To-Entity T...	Relation. Type	Cardinalit
	BUKRS	PBUKR	POSID	Referencing ▼	1 : N
	CON_AREA	PKOKR	POSID	Referencing ▼	1 : N
	WERKS_D	WERKS_W	POSID	Referencing ▼	1 : N

Save and Activate

Following the steps as previous select the visualize data model button from the tool bar.

Your data model should appear as shown.

Data Model		Name	Data Element	Referenced
ZX		Project Definition				
PSPID		WBS Elements				
POSID		WBS element	E...	<input checked="" type="checkbox"/>	PS_POSID	
		Applicant no.	A...	<input type="checkbox"/>	PS_ASTNR	
		Acct asst elem.	A...	<input type="checkbox"/>	PS_BELKZ	
		Billing Element	A...	<input type="checkbox"/>	PS_FAKKZ	
		Company code	A...	<input type="checkbox"/>	PS_VBUKR	BUKRS
		Controlling area	A...	<input type="checkbox"/>	PS_VKOKR	CON_AREA
		Planning Element	A...	<input type="checkbox"/>	PS_PLAKZ	
		Description	A...	<input type="checkbox"/>	PS_POST1	
		Project Type	A...	<input type="checkbox"/>	PS_PRART	
		Project definition	A...	<input checked="" type="checkbox"/>	PS_PSPID	PSPID
		Priority	A...	<input type="checkbox"/>	NW_PRI0	
		WBS element	A...	<input type="checkbox"/>	PS_POSID	
		Description (lo...	A...	<input type="checkbox"/>	USMD_TXTLG	
		No. of Person ...	A...	<input type="checkbox"/>	PS_VERNR	
		Plant	A...	<input type="checkbox"/>	WERKS_D	WERKS_D

Representing the Work Breakdown Structure Hierarchically

Before we begin, it's important to emphasize that WBSs and WBS elements are different. A WBS is a hierarchical representation of WBS elements, whereas the WBS element represents a work package in a WBS. Entity PSPID represents the project definition, and entity POSID represents the WBS element and not the WBS.

Hierarchy leading entity

The hierarchy leading entity is the type 1 entity in which the hierarchy setting is activated. In our example, we'll configure entity POSID as the leading entity.

Return to our list of entities and select POSID.

Then select the details button from the toolbar highlight in our picture.

Display View "Entity Types": Overview

Visualize Data Model

Dialog Structure

- Inactive Data Models
 - Entity Types
 - Attributes
 - Business Object T
 - Entity Types for H
 - Hierarchy Attrit
 - Hierarchy Attrit
 - Relationships
 - Fields of Foreign K
 - Reuse Active Areas

Data Model ZX

Entity Type	Storage/Use Type	Val.Entity	Data Element
BUKRS	Not Changeable via MDG; No ...	No Edition	PS_VBUKR
CON_AREA	Not Changeable via MDG; No ...	No Edition	PS_VKOKR
DATBI	Not Changeable via MDG; No ...	No Edition	DATBI
POSID	Changeable via Change Reque...	No Edition	PS_POSID
PRCTR	Not Changeable via MDG; No ...	No Edition	PRCTR

Change the hierarchy type to 'Yes – Not Version-Dependent/ Synchronised'

Data Model ZX

Entity Type POSID

Entity Types

Storage/Use Type Changeable via Change Request; Generated Database Tables

Validity / Entity No Edition

Data Element PS_POSID

Is Hry Type Yes - Not Version-Dependent / Synchronized

Validity / Hierarchy No Edition

Key Assignment Key Cannot Be Changed; No Internal Key Assignment

☒ Language-Dep. Texts

Next we will return to our entity list and select POSID.

This time we will select Hierarchy attributes

Display View "Entity Types": Overview

Visualize Data Model

Dialog Structure

- Inactive Data Models
 - Entity Types
 - Attributes
 - Business Object T
 - Entity Types for H
 - Hierarchy Attrit
 - Hierarchy Attrit
 - Relationships
 - Fields of Foreign K
 - Reuse Active Areas

Data Model ZX

Entity Type	Storage/Use Type	Val.Entity	Data Element
BUKRS	Not Changeable via MDG; No ...	No Edition	PS_VBUKR
CON_AREA	Not Changeable via MDG; No ...	No Edition	PS_VKOKR
DATBI	Not Changeable via MDG; No ...	No Edition	DATBI
POSID	Changeable via Change Reque...	No Edition	PS_POSID
PRCTR	Not Changeable via MDG; No ...	No Edition	PRCTR

Add 'PSPID' and select 'Hierarchy Name' as Entity Use.

Change View "Entity Types for Hierarchies": Overview

New Entries Visualize Data Model

Dialog Structure

- Inactive Data Models
 - Entity Types
 - Attributes
 - Business Object T
 - Entity Types for H
 - Hierarchy Attrit
 - Hierarchy Attrit

Data Model ZX

Entity Type POSID

Ent.Type of Node	Use
PSPID	Hierarchy Name

Creating a Custom User Interface

This section introduces the concept of building the MDG application UI using the SAP MDG custom object UI framework.

User Interface Framework

The major building blocks of the Custom object UI framework are as follows:

User Interface:

The UI technology for Custom Object UI is Web Dynpro-based Floorplan Manager. Floorplan manager enforces consistency in the UI and compliance with the UI guidelines. By providing generic UI building blocks (GUIBB) and predefined floorplans.

USMD_OVP_GEN is the generic Web Dynpro application delivered by SAP to valid the UI for custom object applications.

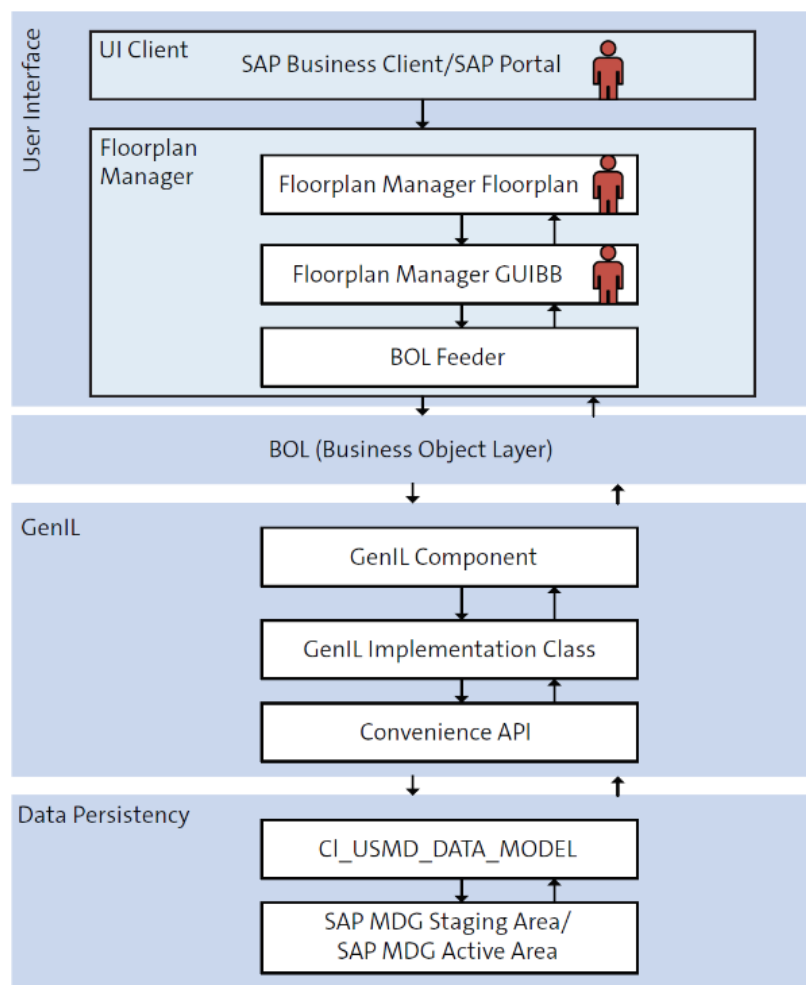
GenIL and Business Object Layer (BOL):

The purpose of GenIL is to provide uniform access to the underlying data persistency layer. It encapsulates business object-specific implementation and provides a uniform interface to access data from the persistency layer.

The major building blocks of the custom object UI framework are shown in the image to the right.

This is the runtime architecture and building blocks of the Custom Object UI Framework.

The loose coupling of the various building blocks guarantees minimum disruption. For example, if the SAP OData framework has a BOL adapter, then we can easily replace the Floorplan Manager with SAP Fiori without changing the code in the underlying GenIL and BOL framework.



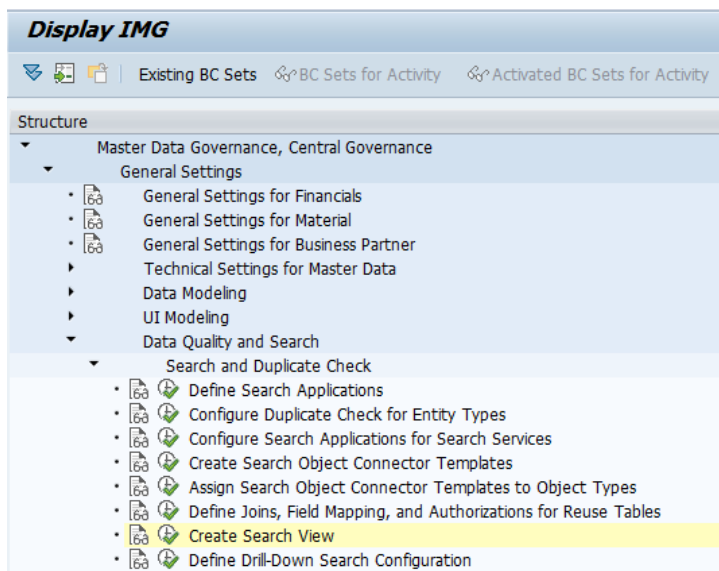
Develop the Search User Interface

SAP Master Data Governance provides a generic search application called USMD_SEARCH, which can be used to search any data model and type 1 entity using any search provider

Our first step toward enabling the generic search UI is to create a search help.

Run Transaction MDGIMG
General Settings →
Data Quality and Search →
Search and Duplicate Check
→
Create Search View.

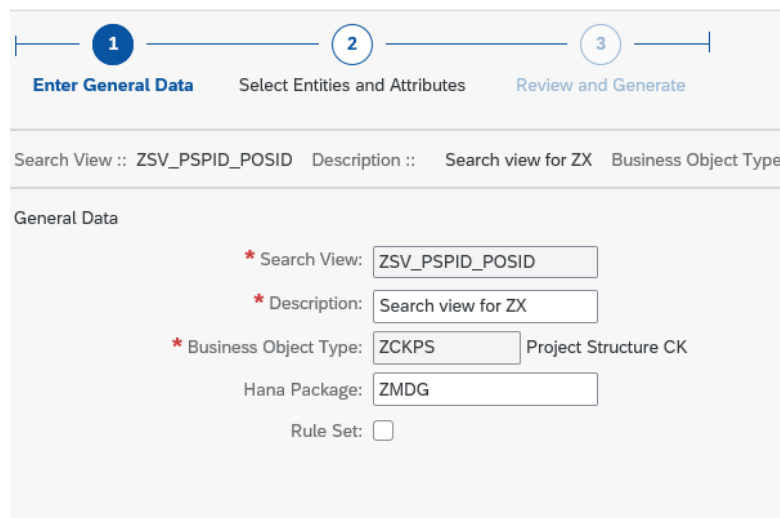
Then select new to create a new search help.



Fill out the general data as shown.

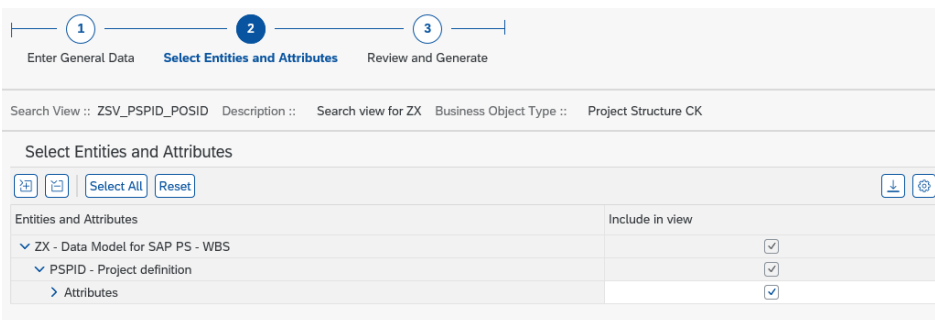
Insert a name in the Search View tab.
A description
Input the business Object we created.

Then select next



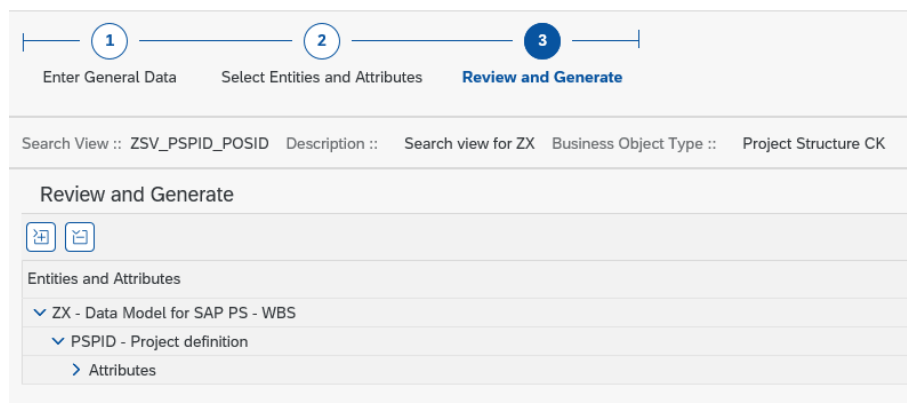
Open the PSPID tab and ensure that attributes is checked to be included in view.

Select Next



Check all entities and attributes are included.

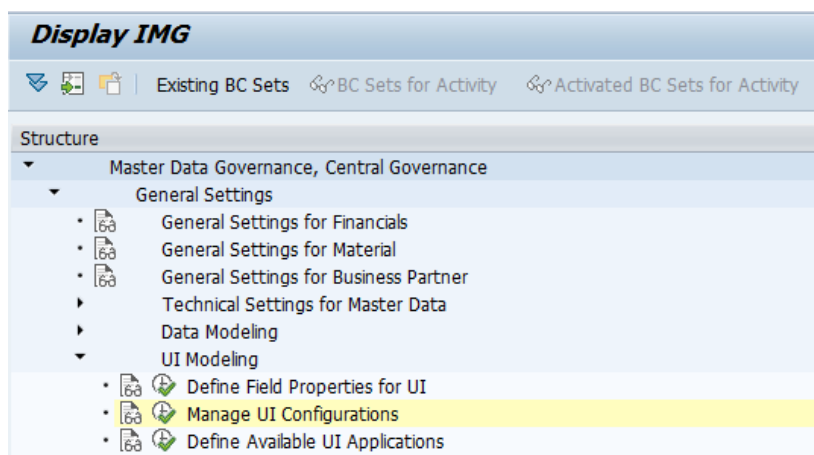
If so, click save.
You will now find your Search view among the list of MDG search views.



Open the Manage UI Configurations Screen..

General Settings →
UI Modeling →
Manage UI Configurations.

This IMG activity is used to manage all the object maintenance-related UIs



We are going to create a new generic search application.

Select the application configuration
USMD_SEARCH_TEMPLATE
and click the copy button

SAP Manage UI Configurations			
Copy			
View: [Standard View] Print Version Export			
Data Model	Application	Application Configuration	UI Configuration
<input type="radio"/>	BS_OVP_CC	BS_OVP_CC	BS_CC_OVP
<input type="radio"/>	USMD_ENTITY_VALUE2	USMD_ENTITY_VALUE2	USMD_ENTITY_VALUE2_OIF
<input type="radio"/>	USMD_OVP_GEN	USMD_OVP_GEN_TEMPLATE	USMD_GEN_OVP_TEMPLATE
<input checked="" type="radio"/>	USMD_SEARCH	USMD_SEARCH_TEMPLATE	USMD_SEARCH_OVP_TEMPLATE
<input type="radio"/>	USMD_SEARCH	USMD_SEARCH_TEMPLATE_02	USMD_SEARCH_OVP_TEMPLATE_02

First thing we will do is change the affixes. To do this select Change Affixes. We will input our data model here ZX.

Select Ok

Change Affixes of Target Configuration IDs

Prefix:

Suffix:

OK


Cancel

Next we can rename the Target configuration, by removing the template.





We can also untick the copy box for Overview Page Floorplan.

Select Deep Copy.

SAP Floorplan Manager: Application Hierarchy Browser - Application: USMD_SEARCH

Browser Mode **Deep-Copy Mode** 

Application Hierarchy: Configuration Level

   **Start Deep-Copy** 

Hierarchy	Copy	Component	Interface...	Configuration ID	Target Configuration ID
Application Configuration	<input checked="" type="checkbox"/>			USMD_SEARCH_TEMPLATE	ZX_USMD_SEARCH
Overview Page Floorplan	<input type="checkbox"/>	FPM_OVP_COM...		USMD_SEARCH_OVP_TEMPLATE	
Search	<input type="checkbox"/>				
SECTION_1					
FPM_SEARCH_UIBB	<input type="checkbox"/>	FPM_SEARCH...	SEARCH...	USMD_SEARCH_DQUERY_TEMP...	
FPM_LIST_UIBB_ATS	<input type="checkbox"/>	FPM_LIST_UIBB...	LIST_WI...	USMD_SEARCH_RESULT_TEMP...	









Return to the list of UI configurations, Find the UI configuration we just created.

Then click on the application Configuration link ZX_USMD_SEARCH.

SAP Manage UI Configurations

Copy

View: [Standard View] Print Version Export

	Data Model	Application	Application Configuration	UI Configuration
		BS_OVP_CC	BS_OVP_CC	BS_CC_OVP
		USMD_ENTITY_VALUE2	USMD_ENTITY_VALUE2	USMD_ENTITY_VALUE2_OIF
		USMD_OVP_GEN	USMD_OVP_GEN_TEMPLATE	USMD_GEN_OVP_TEMPLATE
		USMD_SEARCH	USMD_SEARCH_TEMPLATE	USMD_SEARCH_OVP_TEMPLATE
		USMD_SEARCH	USMD_SEARCH_TEMPLATE_02	USMD_SEARCH_OVP_TEMPLATE_02
		USMD_SEARCH	ZX_USMD_SEARCH	USMD_SEARCH_OVP_TEMPLATE
	OG	MDGF_OVP_GEN	MDGF_OG_OVP_BDC	MDGF_OG_BDC_OVP
	OG	MDGF_OVP_GEN	MDGF_OG_OVP_BDCSET	MDGF_OG_BDCSET_OVP

Select the pencil icon in the top left corner to edit.

Then fill in the following values.

USMD_HIERARCHY):	<input type="text"/>
(USMD_MODEL):	<input type="text"/>
(USMD_OTC):	<input type="text" value="ZCKPS"/>
(USMD_PROCESS):	<input type="text"/>
CH_EDITION_MODE):	<input type="text"/>
MD_SEARCH_HELP):	<input type="text" value="ZSV_PSPID_POSID"/>
ID_SEARCH_MODE):	<input type="text" value="HA"/>

Click Save and Exit

One very important step that needs to be performed after copying the generic search template configuration is to create the communicator (MDG_BS_GOV_COMMUNICATOR) configuration. The name of the communicator configuration and the search UI application must be the same; if they aren't, then the search UI application won't be rendered correctly. The communicator is responsible for rendering the search criteria and search results areas based on the search help configuration.

Now that we made changes you can see our UI configuration has our data model assigned to it.

Click on details.

<input type="radio"/>	Y0	USMD_OVP_GEN	Y0_OVP_COUNTRY
<input type="radio"/>	Y0	USMD_OVP_GEN	Y0_OVP_COUNTRY_RS
<input type="radio"/>	Z0	USMD_ENTITY_VALUE2	ZZ0_USMD_ENTITY_VALUE2
<input type="radio"/>	ZX	USMD_SEARCH	ZX_USMD_SEARCH

Simply ensure that your component and Configuration is set as is.

On the next screen, change nothing, and click **Save**.

Delete Copy Other Functions ▾
0 Hi, CHRIS!

Component Configuration
?

Component Name:

* Configuration ID:

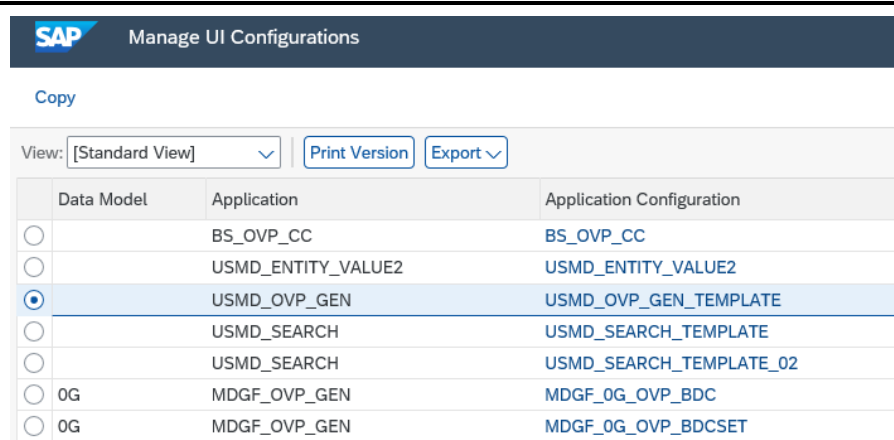
[Show more](#)

Develop a Single Object Maintenance User Interface

In the previous section, we created a search UI; now we'll proceed to create a Single Object Maintenance UI for the project definition and WBS hierarchy.

Return to the list of UI configurations and find the USMD_OVP_GEN Config.

Select and click copy.



The screenshot shows the 'Manage UI Configurations' interface. At the top, there's a 'Copy' button. Below it, a 'View: [Standard View]' dropdown and 'Print Version' and 'Export' buttons. A table lists configurations with columns 'Data Model', 'Application', and 'Application Configuration'. The row for 'USMD_OVP_GEN' is selected, showing 'USMD_OVP_GEN_TEMPLATE' as the application configuration.


	Data Model	Application	Application Configuration
<input type="radio"/>		BS_OVP_CC	BS_OVP_CC
<input type="radio"/>		USMD_ENTITY_VALUE2	USMD_ENTITY_VALUE2
<input checked="" type="radio"/>		USMD_OVP_GEN	USMD_OVP_GEN_TEMPLATE
<input type="radio"/>		USMD_SEARCH	USMD_SEARCH_TEMPLATE
<input type="radio"/>		USMD_SEARCH	USMD_SEARCH_TEMPLATE_02
<input type="radio"/>	OG	MDGF_OVP_GEN	MDGF_OG_OVP_BDC
<input type="radio"/>	OG	MDGF_OVP_GEN	MDGF_OG_OVP_BDCSET

Like before the first thing we should do is change the affixes.

Like before name is as our Data Model. Select OK

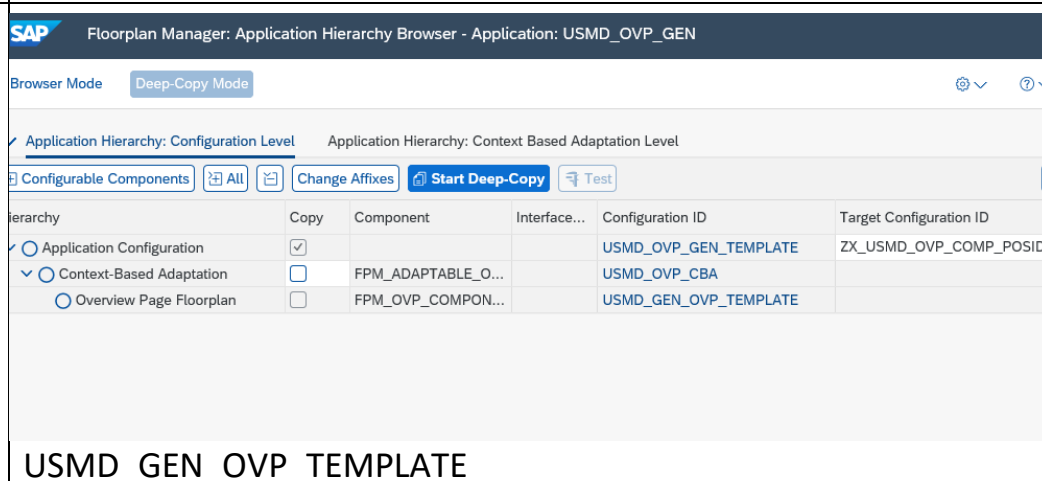
Change Affixes of Target Configuration IDs

Prefix:

Suffix: 

Like before rename Target configuration ID as shown.

Uncheck the copy tick box for control-based application and start deep copy.



The screenshot shows the 'Floorplan Manager: Application Hierarchy Browser' for application 'USMD_OVP_GEN'. It has tabs for 'Browser Mode' and 'Deep-Copy Mode'. Below are buttons for 'Configurable Components', 'All', 'Change Affixes', 'Start Deep-Copy', and 'Test'. A table shows the hierarchy with columns 'Hierarchy', 'Copy', 'Component', 'Interface...', 'Configuration ID', and 'Target Configuration ID'. The 'Application Configuration' row is checked, and its target ID is 'ZX_USMD_OVP_COMP_POSID'. Below the table, the text 'USMD_GEN_OVP_TEMPLATE' is displayed.

Hierarchy	Copy	Component	Interface...	Configuration ID	Target Configuration ID
<input checked="" type="radio"/> Application Configuration	<input checked="" type="checkbox"/>			USMD_OVP_GEN_TEMPLATE	ZX_USMD_OVP_COMP_POSID
<input checked="" type="radio"/> Context-Based Adaptation	<input type="checkbox"/>	FPM_ADAPTABLE_O...		USMD_OVP_CBA	
<input type="radio"/> Overview Page Floorplan	<input type="checkbox"/>	FPM_OVP_COMPON...		USMD_GEN_OVP_TEMPLATE	

USMD_GEN_OVP_TEMPLATE

Return to the UI configurations list and select the hyperlink for your UI
ZX_USMD_OVP_COMP_POSID

Change the USM_OTC to suit your Business Object

(BCV_CONTEXT_KEY):

(CRTYPE):

(FPM_IGNORE_WIRE_SOURCE):

(FPM_PAGING_ACTIVE):

(MDG_HC_COLOR_SAVED):

(MDG_HC_COLOR_UNSAVED):

(MDG_HC_DISABLE):

(USMD_OTC): ⓘ

(USMD_SEARCH_EDITION_MODE):

On the same page at the top we will see 'Component Usage'.

Click on the configuration name
USMD_GEN_OVP_TEMPLATE.

File ▾ Edit ▾ View ▾ Other Functions ▾ Hi, CHRIS



Application Configuration: ZX_USMD_OVP_COMP_POSID

Assign Configuration Name

	Component Usage	Component	Implementation	Configuration Name
<input checked="" type="radio"/>	▼ USMD_OVP_GEN	FPM_ADAPTABLE_OVP	FPM_ADAPTABLE_OVP	USMD_OVP_CBA
<input type="radio"/>	OVP	FPM_OVP_COMPONENT	FPM_OVP_COMPONENT	USMD_GEN_OVP_TEMPLATE

On your new page, in the left column rename the title and page Id to a name of your choosing

Navigation Repositories

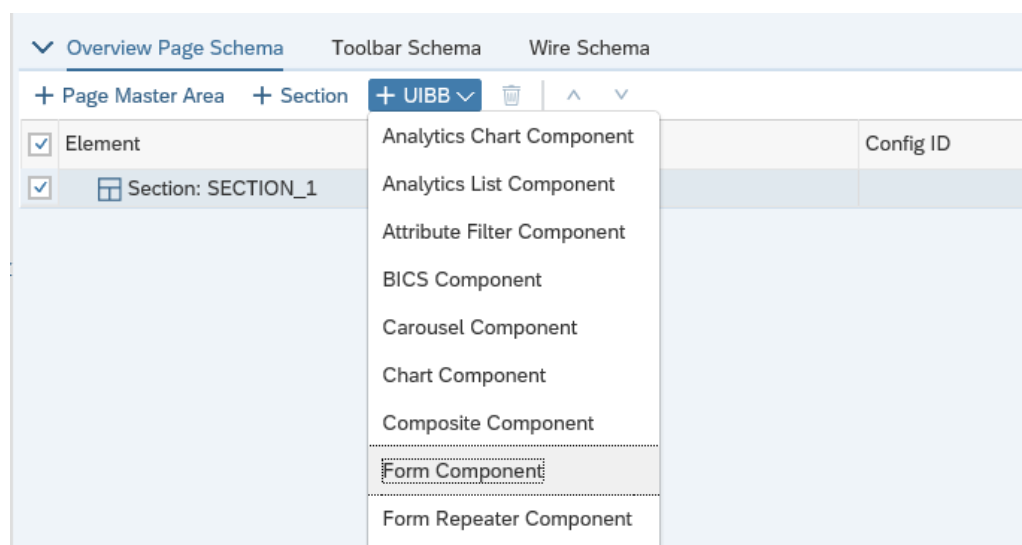
+ ▼   | ^ ▼ 

Page Type	Page Id	Title
Main Page	MAIN	Process Projects

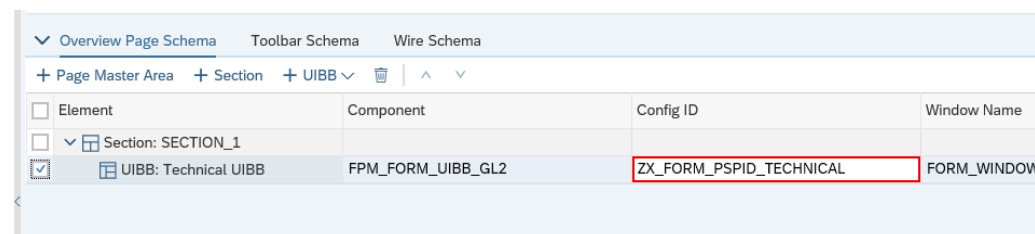
Configure the technical UIBB.

We start the UI configuration of the main page by creating a technical UIBB. The technical UIBB is never shown in the UI, but it participates in the Floorplan Manager event loop. This UIBB will be the root UIBB in the wire schema. The UIBB will just contain the key fields of entity UIBB

In the centre of your page
Select the section button
then select UIBB and
choose Form Component.



When opened fill in the
details as shown



Element	Component	Config ID	Window Name
Section: SECTION_1			
UIBB: Technical UIBB	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID_TECHNICAL	FORM_WINDOW

Ensure you fill out the
fields in the right column
also.

The config ID must match
the name of the config ID
in the previous step.

We have enabled the
Hidden Element attribute
that makes this UIBB
hidden but still
participates in the event
loop.

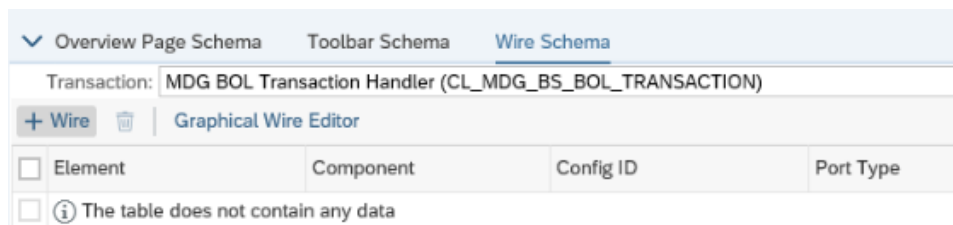
Standard Attributes of UIBB: Technical UIBB

* Component:	FPM_FORM_UIBB_GL2
* Window Name:	FORM_WINDOW
Config ID:	ZX_FORM_PSPID_TECHNICAL
Instance ID:	
Column:	1
Sequence Index:	1
Hidden Element:	Hidden but Processed in Eve...
Rendering Type:	With Panel
Collapsed:	<input type="checkbox"/>
Title:	Technical UIBB
Tooltip:	
Image:	
Padding:	Automatic (Default)
Explanation Text:	

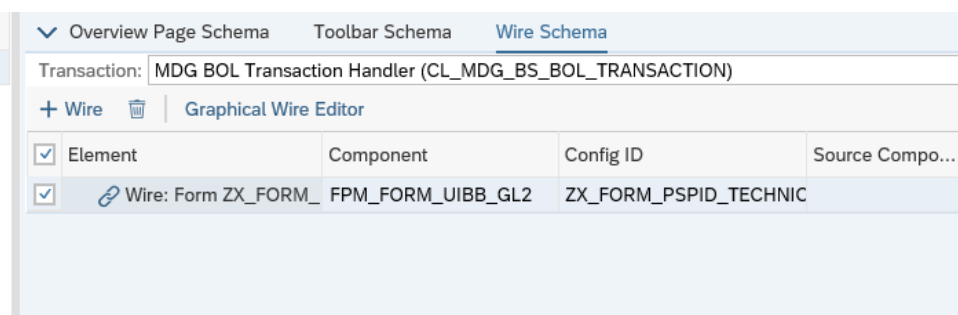
Returning to the centre column,

Change the view from Page Schema to Wire Schema.

Here we will create a wire, which has the technical UIBB as the target UIBB but doesn't have a source UIBB. This makes it the root UIBB, which can be instantiated independently. Select the + Wire Button

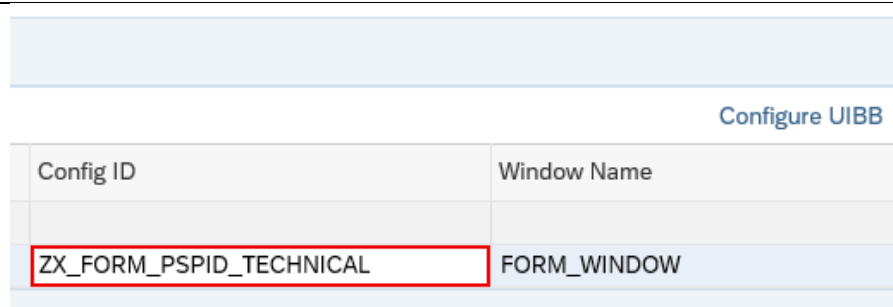


The component should automatically be filled, it is important you fill out the config ID with the same name in the previous steps.

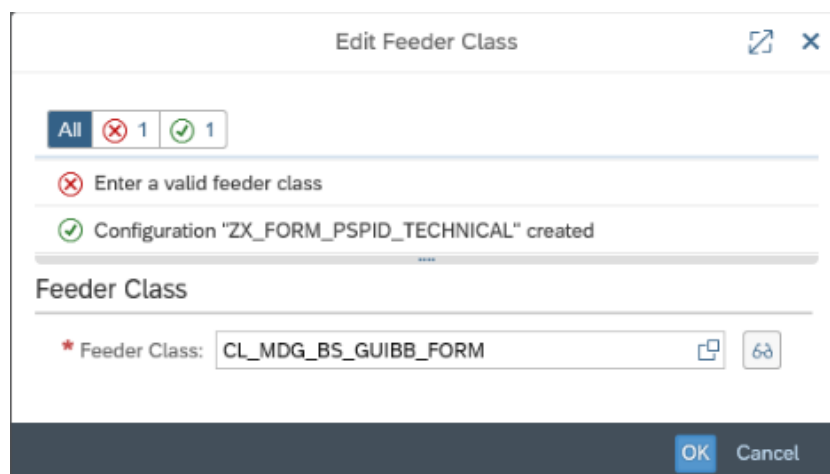


Save your changes

Return to the Page Schema view and select configure UIBB in the top right corner.

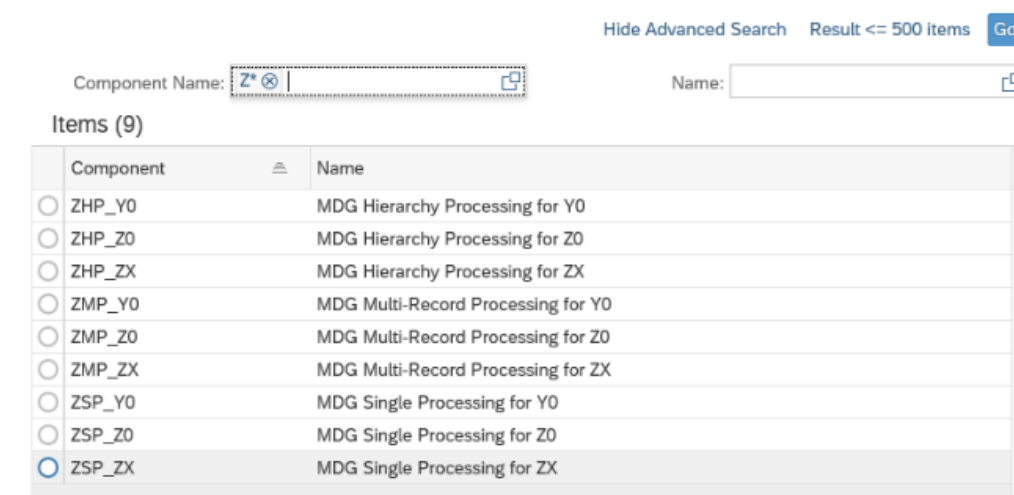


When you come to the feeder screen. Insert the feeder class as shown and click ok



Dialog box titled "Edit Feeder Class". It shows a status bar with "All" (selected), "1" (with a red X), and "1" (with a green checkmark). Below this, there are two messages: "Enter a valid feeder class" (with a red X) and "Configuration 'ZX_FORM_PSPID_TECHNICAL' created" (with a green checkmark). The "Feeder Class" section has a text field labeled "* Feeder Class:" containing the value "CL_MDG_BS_GUIBB_FORM". At the bottom right are "OK" and "Cancel" buttons.

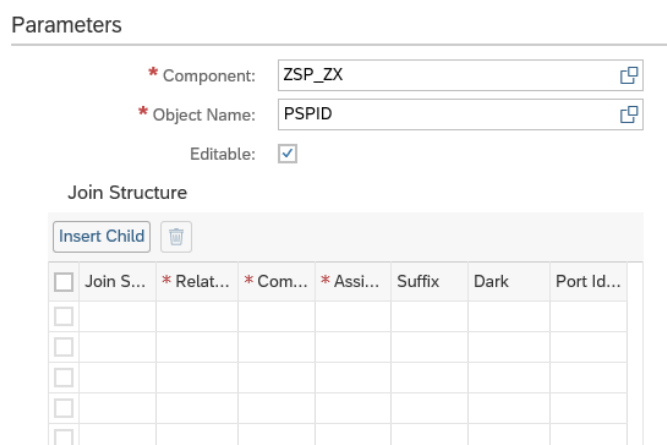
Open the search in the component parameter and select 'ZSP_ZX' as shown



Search results for Component Name: "Z*". The results are displayed in a table with 9 items. The selected item is "ZSP_ZX".

Component	Name
<input type="radio"/> ZHP_Y0	MDG Hierarchy Processing for Y0
<input type="radio"/> ZHP_Z0	MDG Hierarchy Processing for Z0
<input type="radio"/> ZHP_ZX	MDG Hierarchy Processing for ZX
<input type="radio"/> ZMP_Y0	MDG Multi-Record Processing for Y0
<input type="radio"/> ZMP_Z0	MDG Multi-Record Processing for Z0
<input type="radio"/> ZMP_ZX	MDG Multi-Record Processing for ZX
<input type="radio"/> ZSP_Y0	MDG Single Processing for Y0
<input type="radio"/> ZSP_Z0	MDG Single Processing for Z0
<input checked="" type="radio"/> ZSP_ZX	MDG Single Processing for ZX

Insert PSPID as the object name and it is important to always tick the editable box.



Parameters section:

- * Component: ZSP_ZX
- * Object Name: PSPID
- Editable: ☒

Join Structure section:

Buttons: Insert Child, [Trash Icon]





	Join S...	* Relat...	* Com...	* Assi...	Suffix	Dark	Port Id...
<input type="checkbox"/>							
<input type="checkbox"/>							
<input type="checkbox"/>							
<input type="checkbox"/>							
<input type="checkbox"/>							

On the new screen ensure that Project Definition is the only element. You can remove everything else.

	A	B	C	D	E	F	G	H	I	J	K	L
1	PSPID											
2	Project definition: <input type="text"/>											
3												
4												
5												
6												

Make sure to save.
Return by selecting the hyperlink as before:
USMD_GEN_OVP_TEMPLATE

USMD_OVP_GEN: ZX_USMD_OVP_COMP_POSID > OVP: USMD_GEN_OVP_TEMPLATE > Form UIBB: ZX_FORM_PSPID_TECHNICAL

▼ Preview + Element ▼ + Group + Line    




	A	B	C	D	E	F	G	H	I	J	K	L	M
--	---	---	---	---	---	---	---	---	---	---	---	---	---

Configure form UIBB for project definition.

Here we will add the first visible UIBB. This UIBB will be a form and will hold the project definition. We will also configure a wire with port type lead from the technical UIBB to the project definition UIBB.

Create a new UIBB screen as before. Again, create a Form Component UIBB




Overview Page Schema Toolbar Schema Wire Schema

+ Page Master Area + Section + UIBB ▼   

Element	Component	Config ID	Window Name
<input type="checkbox"/> Element	Analytics Chart Component		
<input checked="" type="checkbox"/> Section: SECTION_1	Analytics List Component		
<input type="checkbox"/> UIBB: Technical UIBB	Attribute Filter Component	IL2	ZX_FORM_PSPID_TECHNICAL
	BICS Component		
	Carousel Component		
	Chart Component		
	Composite Component		
	Form Component		
	Form Repeater Component		

Like before fill in a configuration name.


Overview Page Schema Toolbar Schema Wire Schema


+ Page Master Area + Section + UIBB ▼   


Element	Component	Config ID
<input type="checkbox"/> Section: SECTION_1		
<input type="checkbox"/> UIBB: Technical UIBB	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID_TECHNICAL
<input checked="" type="checkbox"/> UIBB	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID

Entering the details in the right column ensure we write the configuration name the same. Note this time we have chosen this as visible

Standard Attributes of UIBB

* Component: FPM_FORM_UIBB_GL2 


* Window Name: FORM_WINDOW 


Config ID: ZX_FORM_PSPID 

Instance ID:

Column:

Sequence Index:

Hidden Element: Visible 

Rendering Type: With Panel 


Collapsed: ☐

Title: Project definition

Like before Create a wire and fill out the config ID same as last step.

The source config name will be the name of the first wire we created
ZX_FORM_PSPID_TECHNICAL

Change port to lead Selection


Overview Page Schema Toolbar Schema Wire Schema						
Transaction: MDG BOL Transaction Handler (CL_MDG_BS_BOL_TRANSACTION)						
+ Wire  Graphical Wire Editor						
Element	Component	Config ID	Source Component	Source Config Name	Port Type	
<input type="checkbox"/> Wire: Form ZX_FORM, FPM_FORM_UIBB_GL2	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID_TECHNICAL				
<input checked="" type="checkbox"/> Wire: Form ZX_FORM, FPM_FORM_UIBB_GL2	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID_TECH	Lead Sele	


In the right column ensure the details are filled out as the same.


Input the port Identifier and
Connector class:
CL_FPM_CONNECTOR_BOL_IDENTITY


Attributes General Settings


Standard Attributes of Wire: Form ZX_FORM_PSPID


* Component: FPM_FORM_UIBB_GL2 


Config ID: ZX_FORM_PSPID 


Instance ID: 


Source Component: FPM_FORM_UIBB_GL2 

Source Config Na...: ZX_FORM_PSPID_TECHNICAL 

Src Inst. ID: 

Port Type: Lead Selection 

Port Identifier: STANDARD 

* Connector Class: CL_FPM_CONNECTOR_BOL_I 

Save Changes

Like before select the Configure UIBB. We will be using the same Feeder class in this case.

Edit Feeder Class ✕

All
✕ 1
✓ 1

✕ Enter a valid feeder class

✓ Configuration "ZX_FORM_PSPID" created

Feeder Class

* Feeder Class: ✕

OK
Cancel

Input the details the same as the previous UIBB.

Feeder Class

Feeder Class: CL_MDG_BS_GUIBB_FORM BOL feeder class GUIBB form for MDG

Parameters

* Component: ✕

* Object Name: ✕

Editable: ☒

This time we can have more elements.

Arrange your elements as shown in the picture.

▼ Preview + Element ▼ + Group + Line ✕









	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	PSPID															
2	Project definition: <input type="text"/>										Applicant no.: <input type="text" value="00000000"/>					
3	Valid To: <input type="text"/>										Factory Calendar: <input type="text"/>					
4	Start date: <input type="text"/>										Finish date: <input type="text"/>					
5	Description: <input type="text"/>										Profit Center: <input type="text"/>					
6	Project Profile: <input type="text"/>										Project Currency: <input type="text"/>					
7	Location: <input type="text"/>										Description (long text): <input type="text"/>					
8	Changed On: <input type="text"/>										Changed By: <input type="text"/>					
9	Created On: <input type="text"/>										Created By: <input type="text"/>					
10	Company code: <input type="text"/>										No. of Person Resp.: <input type="text" value="00000000"/>					
11	Business area: <input type="text"/>										Controlling area: <input type="text"/>					
12	Plant: <input type="text"/>															
13																

Save and return as previous






Configure the list UIBB for language-dependent texts

The project definition (PSPID) has language-dependent texts activated, allowing the description for the project definition to be maintained in multiple languages. To enable this functionality in the UI, you need to add a list UIBB.

Create a new UIBB as previous except this time select List component instead of Form. Fill out the config ID and ensure the details in the right column are filled out correctly.

Overview Page Schema Toolbar Schema Wire Schema			
+ Page Master Area + Section + UIBB     Configure UIBB 			
Element	Component	Config ID	Window Name
<input checked="" type="checkbox"/> Section: SECTION_1			
<input type="checkbox"/>  UIBB: Technical UIBB	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID_TECHNICAL	FORM_WINDOW
<input type="checkbox"/>  UIBB: Project definition	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID	FORM_WINDOW
<input type="checkbox"/>  UIBB: List	FPM_LIST_UIBB_ATS	ZX_LIST_PSPID_TXT	LIST_WINDOW

Create a new wire as previous. Insert a Config ID and name the other columns as previously done.

Overview Page Schema Toolbar Schema Wire Schema					
Transaction: MDG BOL Transaction Handler (CL_MDG_BS_BOL_TRANSACTION)					
+ Wire  Graphical Wire Editor 					
Element	Component	Config ID	Source Component	Source Config Na...	Port Type
<input type="checkbox"/>  Wire: Form ZX_F FPM_FORM_UIBB_GL ZX_FORM_PSPID_TECHNICAL					
<input type="checkbox"/>  Wire: Form ZX_F FPM_FORM_UIBB_GL ZX_FORM_PSPID			FPM_FORM_UIBB_G ZX_FORM_PSPID_T	Lead Selection	
<input checked="" type="checkbox"/>  Wire: FPM_FORM FPM_FORM_UIBB_AT ZX_LIST_PSPID_TXT			FPM_FORM_UIBB_G ZX_FORM_PSPID_T	Lead Selection	



In the right column fill out the details as shown. Note we are using a different connector class this time.

Make sure to fill out the connector parameters as shown

Attributes	General Settings
Standard Attributes of Wire: FPM_FORM_UIBB_ATS ZX_LIST_PSPID_TXT	
* Component:	FPM_FORM_UIBB_ATS
Config ID:	ZX_LIST_PSPID_TXT
Instance ID:	
Source Component:	FPM_FORM_UIBB_GL2
Source Config Name:	ZX_FORM_PSPID_TECHNICAL
Src Inst. ID:	
Port Type:	Lead Selection
Port Identifier:	STANDARD
* Connector Class:	CL_MDG_BS_CONNECTOR_BOL_REL
Connector Parameters	
Relation Name:	PSPID2DTXTSPIDREL
Creation Mode:	Creation with Default Values (Using Template i...

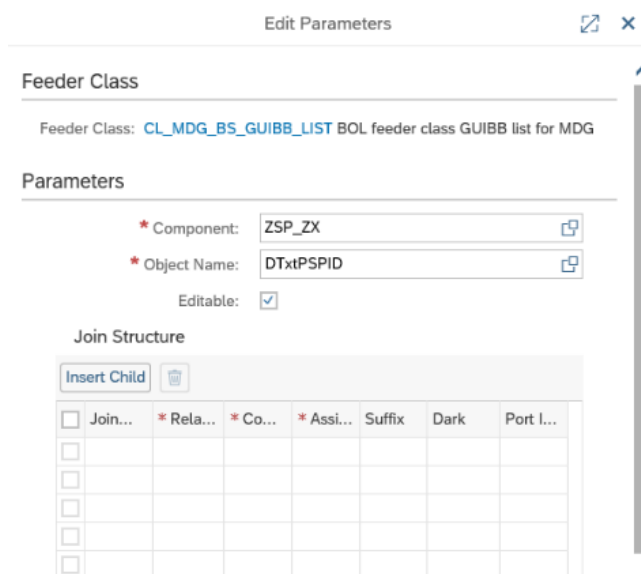
As previous select configure UIBB. This time we are using a different feeder class.

Insert the feeder class shown

Edit Feeder Class	
<div>   2 </div>	
Feeder Class	
* Feeder Class:	CL_MDG_BS_GUIBB_LIST
<div> <div>OK</div> <div>Cancel</div> </div>	

Again select the component as ZSP_ZX.

Insert the object name as DTxPSPID



Edit Parameters

Feeder Class

Feeder Class: CL_MDG_BS_GUIBB_LIST BOL feeder class GUIBB list for MDG

Parameters

* Component: ZSP_ZX

* Object Name: DTxPSPID

Editable: ☒

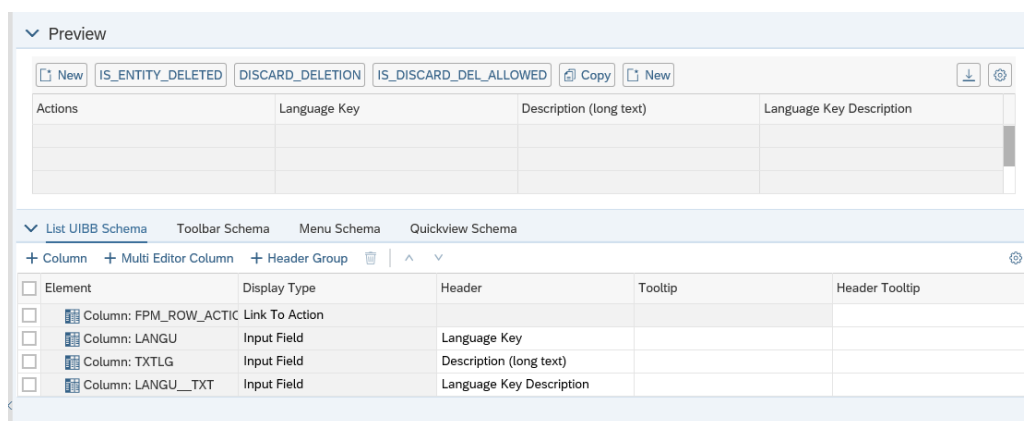
Join Structure

Insert Child

Join...	* Rela...	* Co...	* Assi...	Suffix	Dark	Port I...
<input type="checkbox"/>						
<input type="checkbox"/>						
<input type="checkbox"/>						
<input type="checkbox"/>						
<input type="checkbox"/>						

Arrange the elements as shown in the picture.

Any additional elements can be removed



Preview

IS_ENTITY_DELETED DISCARD_DELETION IS_DISCARD_DEL_ALLOWED Copy New

Actions	Language Key	Description (long text)	Language Key Description

List UIBB Schema Toolbar Schema Menu Schema Quickview Schema

Element	Display Type	Header	Tooltip	Header Tooltip
Column: FPM_ROW_ACTIC Link To Action	Link To Action			
Column: LANGU	Input Field	Language Key		
Column: TXTLG	Input Field	Description (long text)		
Column: LANGU__TXT	Input Field	Language Key Description		



Save and return as previous

Configure the list UIBB for creating and changing WBS elements

A project definition can have multiple WBS elements assigned to it. Each WBS element has its own set of attributes and is assigned to the WBS hierarchy. To meet these requirements, we first create a list UIBB in the main page and then create an edit page; this edit page is tagged as the default edit page for the list UIBB.

Like the previous create a List Component UIBB.

Name as shown and ensure the right column is filled out correctly

Overview Page Schema Toolbar Schema Wire Schema			
+ Page Master Area + Section + UIBB  ^ v Configure UIBB 			
Element	Component	Config ID	Window Name
Section: SECTION_1			
UIBB: Technical UIBB	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID_TECHNICAL	FORM_WINDOW
UIBB: Project definition	FPM_FORM_UIBB_GL2	ZX_FORM_PSPID	FORM_WINDOW
UIBB: List	FPM_LIST_UIBB_ATS	ZX_LIST_PSPID_TXT	LIST_WINDOW
<input checked="" type="checkbox"/> UIBB: WBS Elements	FPM_LIST_UIBB_ATS	ZX_POSID_LIST_WBS	LIST_WINDOW

Create a wire like previous with the same Source component and Source Config name

Overview

Page Schema

Toolbar Schema

Wire Schema

Transaction: MDG BOL Transaction Handler (CL_MDG_BS_BOL_TRANSACTION)









+ Wire

Graphical Wire Editor

<input type="checkbox"/>	Element	Component	Config ID	Source Component	Source Config Na...	Port Type
<input type="checkbox"/>	Wire: Form ZX_F	FPM_FORM_UIBB_GL	ZX_FORM_PSPID_TECHNICAL			
<input type="checkbox"/>	Wire: Form ZX_F	FPM_FORM_UIBB_GL	ZX_FORM_PSPID	FPM_FORM_UIBB_C	ZX_FORM_PSPID_T	Lead Selection
<input type="checkbox"/>	Wire: FPM_FORM	FPM_FORM_UIBB_AT	ZX_LIST_PSPID_TXT	FPM_FORM_UIBB_C	ZX_FORM_PSPID_T	Lead Selection
<input checked="" type="checkbox"/>	Wire: FPM_FORM	FPM_FORM_UIBB_AT	ZX_POSID_LIST_WBS	FPM_FORM_UIBB_C	ZX_FORM_PSPID_T	Lead Selection



In the right column fill out the details as shown. Note we have changed the Connector class again.



The port identifier is the same.


Attributes	General Settings
Standard Attributes of Wire: List ZX_LIST_PSPID_TXT	
* Component:	FPM_LIST_UIBB_ATS 
Config ID:	ZX_LIST_PSPID_TXT 
Instance ID:	
Source Component:	FPM_FORM_UIBB_GL2 
Source Config Name:	ZX_FORM_PSPID_TECHNICAL 
Src Inst. ID:	
Port Type:	Lead Selection
Port Identifier:	STANDARD 
* Connector Class:	CL_MDG_BS_CONNECTOR_BOL_ASSOC 
Connector Parameters	
Relation Name:	PSPID2DtPSPIDRel
Creation Mode:	Creation with Default Values (Using Template ...)


Save and click configure UIBB

Insert the feeder class,
we are using the same
feeder class as previous

Edit Feeder Class  



All  1  1

 Enter a valid feeder class

 Configuration "ZX_LIST_POSID_WBS" created

....

Feeder Class

* Feeder Class:  


OK **Cancel**


Fill out the details as
shown. Select OK

Feeder Class

Feeder Class: BOL feeder class GUIBB list for MDG

Parameters

* Component: 

* Object Name: 

Editable: ☒





In the right Column you
will see an 'Action
Assignment '.

Insert Events:

Show

FPM_CALL_DEFAULT_EDIT_PAGE




Action Assignment



  |  

	FPM Event ID	Image	Label	Hide Label	Tooltip
<input checked="" type="radio"/>	SHOW		Details	<input type="checkbox"/>	
<input type="radio"/>	FPM_CALL_DEFAU...	~Icon/Edit		<input type="checkbox"/>	
<input type="radio"/>	DISCARD_DELETE	~Icon/Undo		<input type="checkbox"/>	Discard Deletion
<input type="radio"/>					

Arrange your UIBB layout
so it appears as shown in
the picture.







Preview







  

Actions	WBS element	Applicant no.	Description	Project definition	WBS element (Inter...
<input type="radio"/>					
<input type="radio"/>					
<input type="radio"/>					

List UIBB Schema **Toolbar Schema** **Menu Schema** **Quickview Schema**

 **Column**  **Multi Editor Column**  **Header Group**  |  

Element	Display Type	Header	Tooltip	Header Tooltip
<input type="checkbox"/>  Column: FPM_ROW_A Link To Action				
<input type="checkbox"/>  Column: POSID	Input Field	WBS element		
<input type="checkbox"/>  Column: ASTNR	Input Field	Applicant no.		
<input checked="" type="checkbox"/>  Column: POST1	Input Field	Description		
<input type="checkbox"/>  Column: PROJ_DEF	Input Field	Project definition		
<input type="checkbox"/>  Column: PS_POSID	Input Field	WBS element (Internal id)		

Save Changes

We have now fully implemented the UI for our custom data model.

Process Modelling

In MDG every change including the creation of master data is done through a change request, which is roughly a carrier of changed to the master data. After change request are initiated, they need to be processed by applying governance rules and collaboration.

The process model provides the required input for change request creation and process of change requests. The process modelling configuration node in SAP MDG is a group of similar configuration activities required to execute the change request process.

For process modelling, the first step is to create a new business activity to tie an action with the business object and then maintain navigation settings of the custom UI application using business activities and actions.

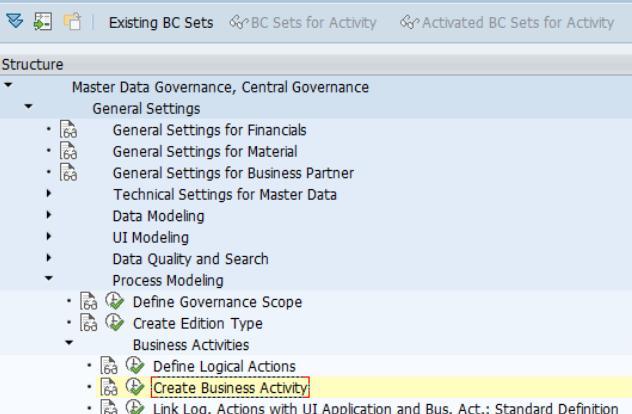
Create a New Business Activity

To design a business activity, you first must understand what kind of actions your intent to perform and on which business object. Each combination of logical action and business object becomes one business activity. In our case, we will create two business activities.

ZBA1 Create project definition and WBS elements.

ZBA2 Change project definition and assign WBS elements to the project hierarchy.

Display IMG



New Entries: Overview of Added Entries



Business Activity: Definition

Bus.Acty	Description (medium text)	D., Description (medium text)	BO Type	Description	Log. Action	Description
ZBA1	Create Project Definition	ZX Data Model for SAP PS - WBS	ZCKPS	Project Structure CK	CREATE	Create
ZBA2	Change Proj Definition	ZX Data Model for SAP PS - WBS	ZCKPS	Project Structure CK	CHANGE	Change

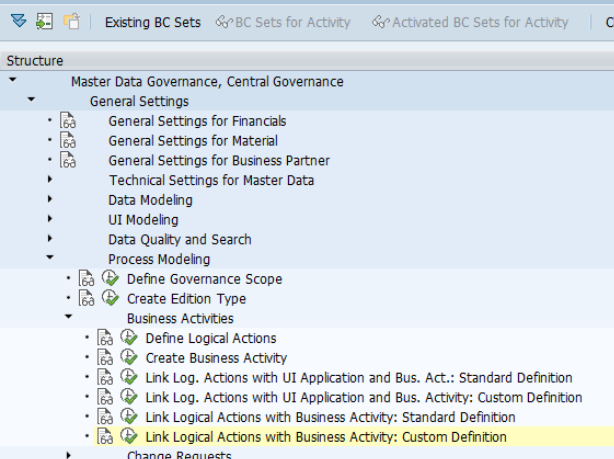
Assign Business Activities and Logical Actions to Business Objects

A logical action represents the operation to be performed on the master data by an actor in the process (e.g., create, change, or delete). Business activities add business context to logical actions by linking them with business objects such as create supplier, change material, and delete account. Business activities are defined by assigning an action, data model, and business object.

In transaction mdgmg:

General settings →
Process Modelling →
Business Activities →
Link Logical Actions with
Business Activity Custom
Definition.

Display IMG



In our case we don't need to create a new action but reuse a SAP delivered action such as change and create. To do this select the two actions with change and create and click copy as

Change View "Business Activity: Determination": Overview

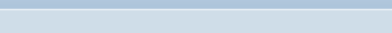


Business Activity: Determination

UI Application Name	UI Configuration	Log. Action	Description	Bu...	Description (medium text)
MDG_BS_GEN_MC_O...	MDG_BS_GEN_MC_OVP_SF	MULTI	Multiple-Reco...	SFCM	Processing of Multiple Airlines
MDG_BS_MAT_OVP	ZBS_MAT_OVP00	CREATE	Create	MAT1	Create Material
USMD_OVP_GEN	USMD_SF_OVP_CARR_02	CHANGE	Change	SFC2	Change Airline
USMD_OVP_GEN	USMD_SF_OVP_CARR_02	CREATE	Create	SFC1	Create Airline

We can no edit those actions to suit out UI Configuration and Business activity as shown.

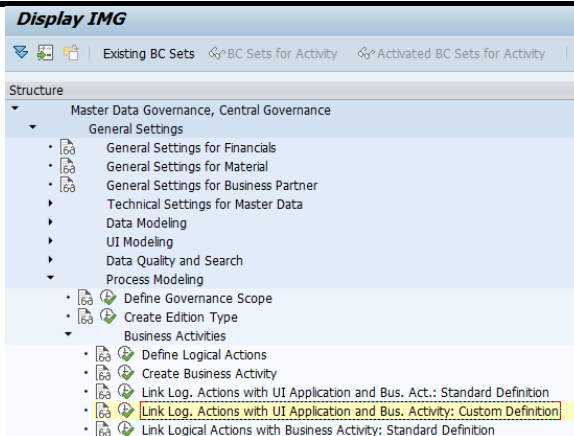
Change View "Business Activity: Determination": Overview of Selected S



Business Activity: Determination

UI Application Name	UI Configuration	Log. Action	Description	Bu...	Description (medium text)
USMD_OVP_GEN	ZX_USMD_OVP_COMP_POSID	CHANGE	Change	ZBA2	Change Proj Definition
USMD_OVP_GEN	ZX_USMD_OVP_COMP_POSID	CREATE	Create	ZBA1	Create Project Definition

Creating cross- application navigation. Firstly like before access this through mdgimng
 General settings →
 Process Modelling →
 Business Activities →
 Link Log Actions with UI Application and Bus.
 Activity Custom Definition.



Select new entries and enter the values shown.

This is the list of configurations created for cross-application navigation between the generic search UI and the custom object UI for actions create, change, and display.

New Entries: Overview of Added Entries

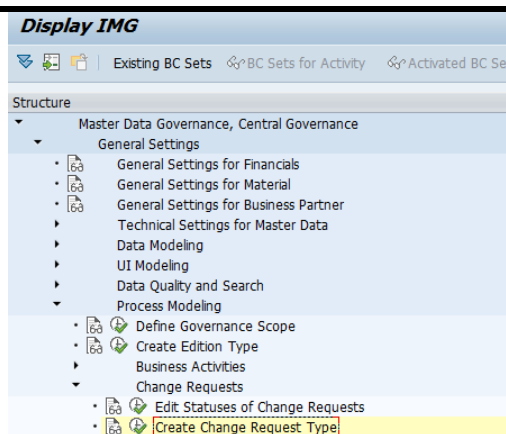
Maintenance Navigation

BO Type	Log. Action	Current UI Application Name	Current UI Configuration	Target UI Application Name	Target UI Configuration	Bus.Acty
ZCKPS	CHANGE	*	*	USMD_OVP_GEN	ZX_USMD_OVP_COMP_POSID	ZBA2
ZCKPS	CHANGE	USMD_SEARCH	ZX_USMD_SEARCH	USMD_OVP_GEN	ZX_USMD_OVP_COMP_POSID	ZBA2
ZCKPS	CREATE	*	*	USMD_OVP_GEN	ZX_USMD_OVP_COMP_POSID	ZBA1
ZCKPS	CREATE	USMD_SEARCH	ZX_USMD_SEARCH	USMD_OVP_GEN	ZX_USMD_OVP_COMP_POSID	ZBA1
ZCKPS	DISPLAY	*	*	USMD_OVP_GEN	ZX_USMD_OVP_COMP_POSID	
ZCKPS	DISPLAY	USMD_SEARCH	ZX_USMD_SEARCH	USMD_OVP_GEN	ZX_USMD_OVP_COMP_POSID	

Create Change Request Type

The change request type is a key characteristic of the change request that determines how a change request is processed. The change request type links a change request to the workflow, data model and business Activity.

Open Create Change request type through the mdgimng transaction as shown



Find the Mat01 Change Request and select copy.

Then change the details to as shown

Change View "Type of Change Request": Overview of Selected Set

Type of Change Request						
Type of Chg. Request	E...	Data Model	Description (medium text)	Main Entity Type	Workflow	Single Obj...
ZXWBSCRE		ZX	Create Projects	PSPID	WS60800086	<input checked="" type="checkbox"/>

We will need to delete the Entity Types select all and then remove.

Change View "Entity Types": Overview

Entity Types			
Entity Type	Scenario	Configuration ID	Op
DRADBASIC			
MATCHGNG			
MATERIAL			
MKALBASIC			

The select new entries and add the two Entity Types shown.

New Entries: Overview of Added Entries

Entity Types					
Entity Type	Scenario	Configuration ID	Optional	Message Output	
PSPID			<input type="checkbox"/>	Standard	
POSID			<input type="checkbox"/>	Standard	
<input checked="" type="checkbox"/>			<input type="checkbox"/>	Standard	
<input checked="" type="checkbox"/>			<input type="checkbox"/>	Standard	
<input checked="" type="checkbox"/>			<input type="checkbox"/>	Standard	

Open business activities and like before remove the activity associated with MAT01 and then add the business activity we created for create.

Change View "Business Activities": Overview

Business Activities	
Bus. Activity	Description (medium text)
ZBA1	Create Project Definition

We need to create a 2nd for change. We can simply copy the Change request type we just created.

Change View "Type of Change Request": Overview

Type of Change Request			
Type of Chg. Request	Edition Type	Data Model	Description (medium text)
ZXWBSCRE		ZX	Create Projects

Change the details to as shown to associate it with changing projects.

Type of Change Request						
Type of Chg. Request	E...	Data Model	Description (medium text)	Main Entity Type	Workflow	Single Obj...
ZXWBSCRG		ZX	Change Projects	PSPID	WS60800086	<input checked="" type="checkbox"/>

Change the business activity to the one we created for change Projects

Dialog Structure

Type of Change Request

Entity Types

Scope on Entity T

Business Activities

Service Level Agree

Type of Chg. Request

ZXWBSCRG

Business Activities

Bus. Activity	Description (medium text)
ZBA2	Change Proj Definition




Save your Changes

Create a workflow


We need to assign a workflow to the change request type. In our case we will assign a rule-based workflow as it is more structured.

Like previously we will be opening this through the mdgimg tab.

Open configure rule-based workflow as shown in the picture.

Display IMG	
 Existing BC Sets  BC Sets for Activity  Activated BC Sets for Activity	
Structure	
<ul style="list-style-type: none"> Master Data Governance, Central Governance <ul style="list-style-type: none"> General Settings <ul style="list-style-type: none"> General Settings for Financials General Settings for Material General Settings for Business Partner Technical Settings for Master Data Data Modeling UI Modeling Data Quality and Search Process Modeling <ul style="list-style-type: none"> Define Governance Scope Create Edition Type Business Activities Change Requests Workflow <ul style="list-style-type: none"> Activate Event Type Linkage Configure Workflow Tasks Define Change Request Actions Define Change Request Step Types and Assign Actions Rule-Based Workflow <ul style="list-style-type: none"> Define Change Request Steps for Rule-Based Workflow Define Service Names for Rule-Based Workflow Configure Rule-Based Workflow 	

Insert the create change request type we made as shown

SAP Process Definition of Rule-Based Workflow	
* Type of Change Request:	ZXWBSCRE  Create Projects

Locate the Decision tables and Open the Non-User decision table.

Insert two rows as shown.

■ Decision Table: DT_NON_USER_AGT_GRP_ZXWBSCRE

[Back](#) |
 [Display](#) |
 [Check](#) |
 [Save](#) |
 [Activate](#) |
 [Delete](#) |
 [More](#)

General

Detail

[Additional Actions](#) |
 [Context Overview](#) |
 [Start Simulation](#)

Table Contents

[+](#) [-](#) [✎](#) [📄](#) [📁](#) |
 [^](#) [v](#) |
 Find: |
 [Next](#) [Previous](#)

<input type="checkbox"/> Condition Alias	Agent Group	Process Pattern
<input type="checkbox"/> =2	001	99 (Complete (Sub-)Workflow)
<input type="checkbox"/> =1	001	05 (Activation (Do Not Bypass Snapshot))

Save and Activate

Next open the Single Value Decision table and fill in the values as shown.

✓ 1 Message [Show I](#)

◆ Decision Table: DT_SINGLE_VAL_ZXWBSCRE

Active version available

[Back](#) |
 [Display](#) |
 [Check](#) |
 [Save](#) |
 [Activate](#) |
 [Delete](#) |
 [More](#)

General

Detail

[Additional Actions](#) |
 [Context Overview](#) |
 [Start Simulation](#)

Table Contents

[+](#) [-](#) [✎](#) [📄](#) [📁](#) |
 [^](#) [v](#) |
 Find: |
 [Next](#) [Previous](#) [Table Settings](#)

<input type="checkbox"/> CR Previous Step	Previous Action	Condition Alias	New Chng. Req. Step	New CR Status
<input type="checkbox"/> =90	=09 (Activate)	1	91	02 (Changes to Be E...
<input checked="" type="checkbox"/> =91	=31 (Activati...	2	99	05 (Final Check Appr...
<input type="checkbox"/>				
<input type="checkbox"/>				

Save and Activate

Creating our CR Wire

Now that we have created our Change Request and assigned a workflow, we must create a link between that and our UI. We have created our UI and our CR although as of now they are not linked and our model wont work. We need to create a CR wire to do this in our UI configurations.

Locate the UI configurations page as we did earlier and find our UI configuration.

Select Details

Then select change

ZX_USMD_OVP_COMP_POSID	Details	USMD_GEN_OVP_TEMPLATE
ZX_USMD_SEARCH	Details	USMD_SEARCH_OVP_TEMPLATE

We need to create a wire to connect our change request with our UI.

Select new → crWires

▼ Component-Defined

You can insert elements using the context menu for table rows

Configuration Context

New	
searchUibbs	
crWires	
▼ settings	

Fill in the details as shown.

As you can see the page Id represents the page we created earlier.

The source config name is our Technical UIBB.

* Page Id:	MAIN	<input checked="" type="checkbox"/> Final
connector:	CL_MDG_BS_CONNECTOR_BOL_CR_REL	<input type="checkbox"/> Final
Port Type:	Lead Selection	<input type="checkbox"/> Final
Port Identifier:	STANDARD	<input type="checkbox"/> Final
Source Component:	FPM_FORM_UIBB_GL2	<input type="checkbox"/> Final
Source Config Name:	ZX_FORM_PSPID_TECHNICAL	<input type="checkbox"/> Final
Src Config Type:	General	<input type="checkbox"/> Final
Src Config Ver:		<input type="checkbox"/> Final
Src Inst. ID:		<input type="checkbox"/> Final

Creating our Role

Roles are used for configuring authorization profiles and menus for users. Roles are directly assigned to the user master and can also be used to configure personalization values for users. The role for menu is used exclusively to configure the menu entries in SAP Business Client.

Search Transaction 'PFCG'.

Enter the role
'SAP_MDGX_MENU_05'

Then select copy in the top left corner.

Role Maintenance

Transactions

Role: SAP_MDGX_MENU_05

Short Description:

Views: Show Documentation

Favorites	Description	Target Sys

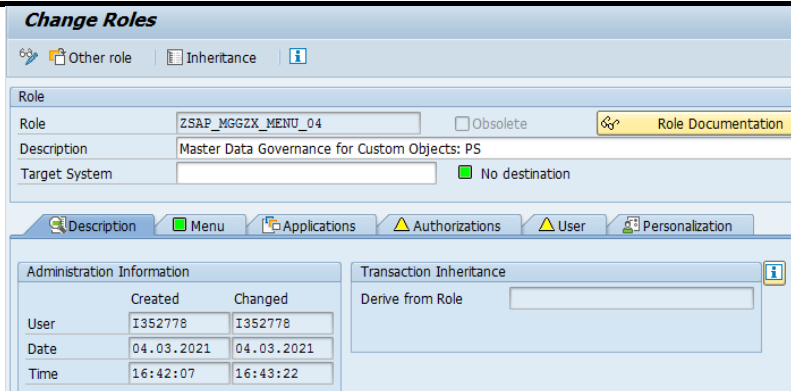
Then enter the name of your role and select ok

Query

Enter Source/Target

from role	SAP_MDGX_MENU_05
to role	ZSAP_MGGZX_MENU_04

Enter a description for your role



Change Roles

Other role | Inheritance | i

Role

Role: ZSAP_MGGZX_MENU_04 ☐ Obsolete [Role Documentation](#)

Description: Master Data Governance for Custom Objects: PS

Target System: ☒ No destination

Description | Menu | Applications | Authorizations | User | Personalization

Administration Information

	Created	Changed
User	I352778	I352778
Date	04.03.2021	04.03.2021
Time	16:42:07	16:43:22

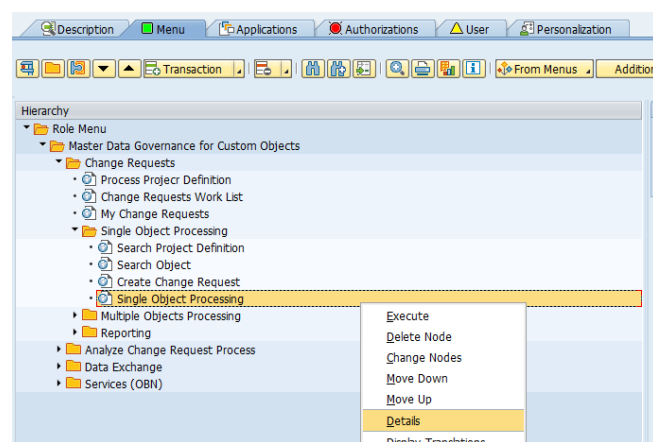
Transaction Inheritance

Derive from Role:

Save

Select the Menu tab and then open single Object Processing

The right click on the single object process tab and Select details.



Description | Menu | Applications | Authorizations | User | Personalization

Transaction

From Menus | Additional

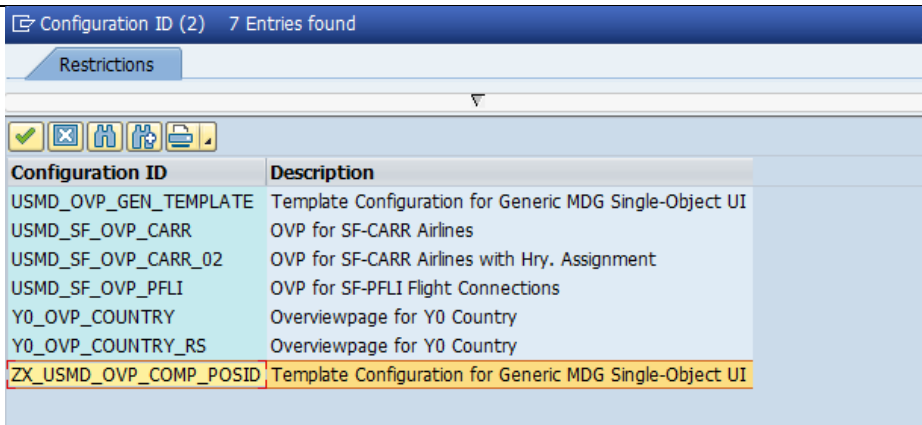
Hierarchy

- Role Menu
 - Master Data Governance for Custom Objects
 - Change Requests
 - Process Project Definition
 - Change Requests Work List
 - My Change Requests
 - Single Object Processing
 - Search Project Definition
 - Search Object
 - Create Change Request
 - Single Object Processing
 - Multiple Objects Processing
 - Reporting
 - Analyze Change Request Process
 - Data Exchange
 - Services (OBN)

Context menu for Single Object Processing:

- Execute
- Delete Node
- Change Nodes
- Move Down
- Move Up
- Details
- Navigation Transitions

Enter the Configuration Id as the ID we created. You can search this to ensure you select the correct one.

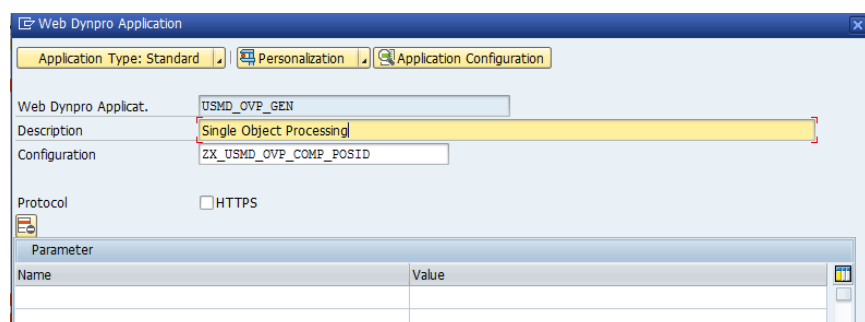


Configuration ID (2) 7 Entries found

Restrictions

Configuration ID	Description
USMD_OVP_GEN_TEMPLATE	Template Configuration for Generic MDG Single-Object UI
USMD_SF_OVP_CARR	OVP for SF-CARR Airlines
USMD_SF_OVP_CARR_02	OVP for SF-CARR Airlines with Hry. Assignment
USMD_SF_OVP_PFLI	OVP for SF-PFLI Flight Connections
Y0_OVP_COUNTRY	Overviewpage for Y0 Country
Y0_OVP_COUNTRY_RS	Overviewpage for Y0 Country
ZX_USMD_OVP_COMP_POSID	Template Configuration for Generic MDG Single-Object UI

You can change the description or leave as is then click ok



Web Dynpro Application

Application Type: Standard | Personalization | Application Configuration

Web Dynpro Applicat.: USMD_OVP_GEN

Description: Single Object Processing

Configuration: ZX_USMD_OVP_COMP_POSID

Protocol: ☐ HTTPS

Parameter

Name	Value

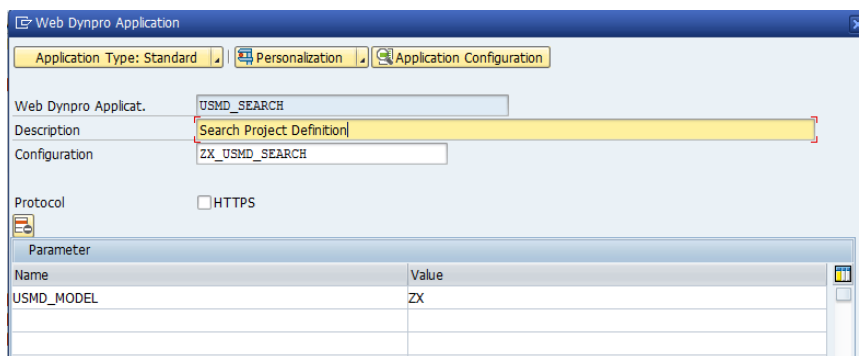
Like before open the Single Object Processing folder, then right click on the Search Object and click details.

Change the configuration to the Search config you created.

You can change your Description or leave as is.

Input your Model as shown.

Save your role.



Name	Value
USMD_MODEL	ZX

Testing our Data Model

Open Transaction NWBC.

Scroll down and select the role we created

[SAP MDG SUPPLIER APP](#)
[SAP MDG WF ADM](#)
[ZSAP_MGGZX_MENU_04](#)

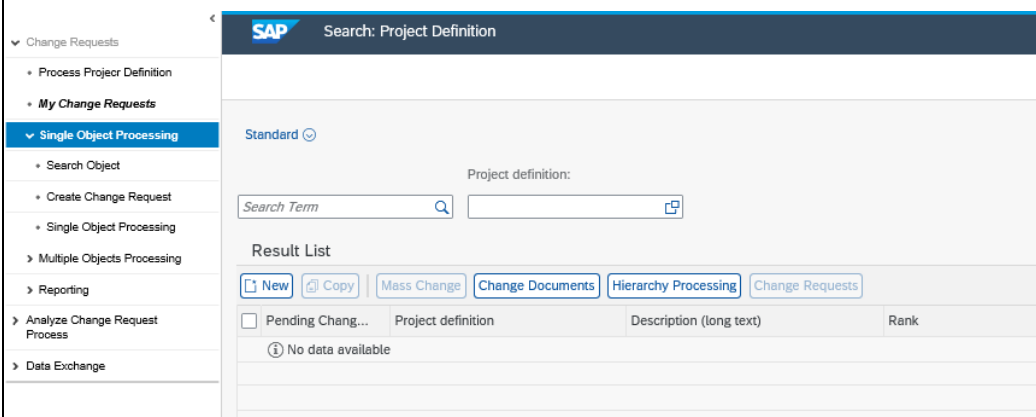
SAP Fiori : Supplier approval

Master Data Governance: Authorizations for Workflow Batch User

Master Data Governance for Custom Objects: PS

Select Single Object processing as you can see we will be brought to our Search screen.

Select new to create a new PS



Pending Chang...	Project definition	Description (long text)	Rank
No data available			

Next we will see the whole UI we created.

The Project definition attributes.

Change Request

Edit

General

Notes

Attachments

General Data

Process Data

Change Request: 197

Status of Change Request: Changes to Be Executed

* Description:

Current Work Item: New Change Request [What's Next](#)

Priority:

Created On/By: 05.03.2021 13:37:29 CHRIS KAVANA...

Due Date:

Type of Change Request: ZXWBSCRE

Reason:

Status of Change Request: 02 Changes to Be Executed

Project definition

Delete

Create

PSPID

* Project definition:

Applicant no.: 00000000

Valid To:

Factory Calendar:

Start date:

Finish date:

Description:

Profit Center:

Project Profile:

Project Currency:

* Location:

Description (long text):

Changed On:

Changed By:

Created On:

Created By:

* Company code:

No. of Person Resp.: 00000000

Business area:

Controlling area:

* Plant:

WBS Elements

Delete

New

Copy

New

POSID

New

IS_ENTITY_DELETED

DISCARD_DELETION

IS_DISCARD_DEL_ALLOWED

DISABLE_HIGHLIGHT_DEL

Actions

WBS element

Applicant no.

Description

No data available

We can additionally see WBS elements option.

