

how.2

Enable mass maintenance of customer fields in material master (MM17).

Author(s):	Filip Stankowski
Created on:	2013-12-12
Component version:	SAP_BASIS 701 (SP 0009)
Document ID:	A001
Summary:	You cannot by default edit custom (ZZ*) fields added to material master data tables (like MARA or MARC) via MM17 transaction. This tutorial explains all steps required to enable this functionality.

Table of Contents

- 1. Before you start 3
- 2. Enhance IDoc MATMAS03 5
- 3. Create implementation for MG_MASS_NEWSEG BAdI 9
- 4. Implement customer exit MGV0000114
- 5. MASS object customizing17
- 6. Field selection group settings.18
- 7. Test your code.19

1. Before you start

This tutorial is based on SAP note 116311 - ALE: Missing material master fields. Please read it before continuing.

Enhancement of MM01/MM02/MM03 transactions is not part of this tutorial so the only thing which has to be prepared before we enable ZZ* field changeability in MM17 transaction is enhancement of one of material master data tables. Let's assume that we need additional ZZZDUMMY field in MARC table. In order to add it you have to:

- 1.1. Start transaction SE11, enter table name in [Database table] field and click [Display] button.

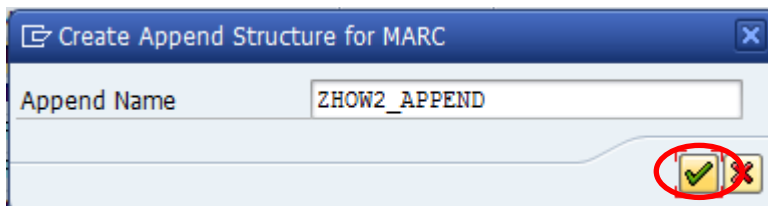
ABAP Dictionary: Initial Screen

Database table
 View
 Data type
 Type Group
 Domain
 Search help
 Lock object

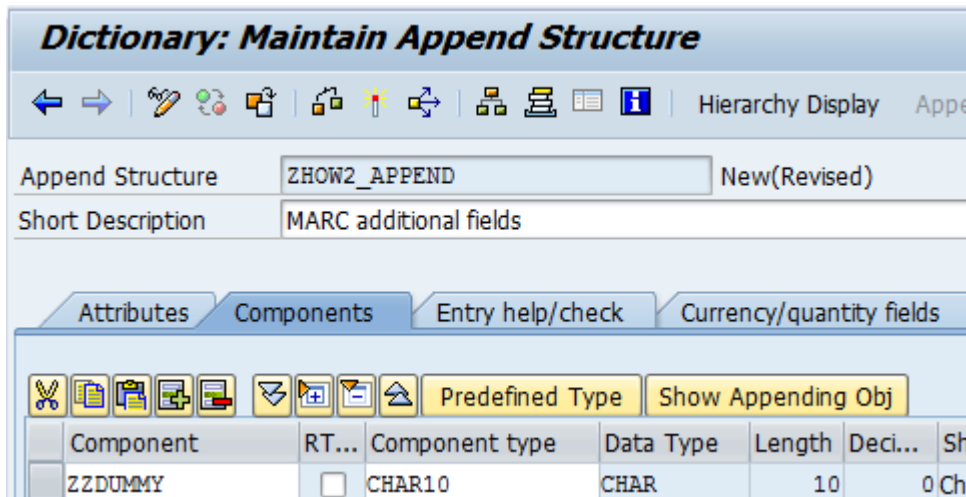
- 1.2. After displaying table definition click on [Append Structure] button (or choose it from [Goto] menu) and click [Create Append] button.

Object Name	Status	Short text
/CWM/MARC	New	CWM Fields for PME
/CWM/MARC_COSTING	New	Catch Weight: Additional Costing Data at Plan
ISH_MARC_APP	New	IS-H MM: Hospital-Specific Fields
IST_MARC	New	IS-T: MARC Additional Fields

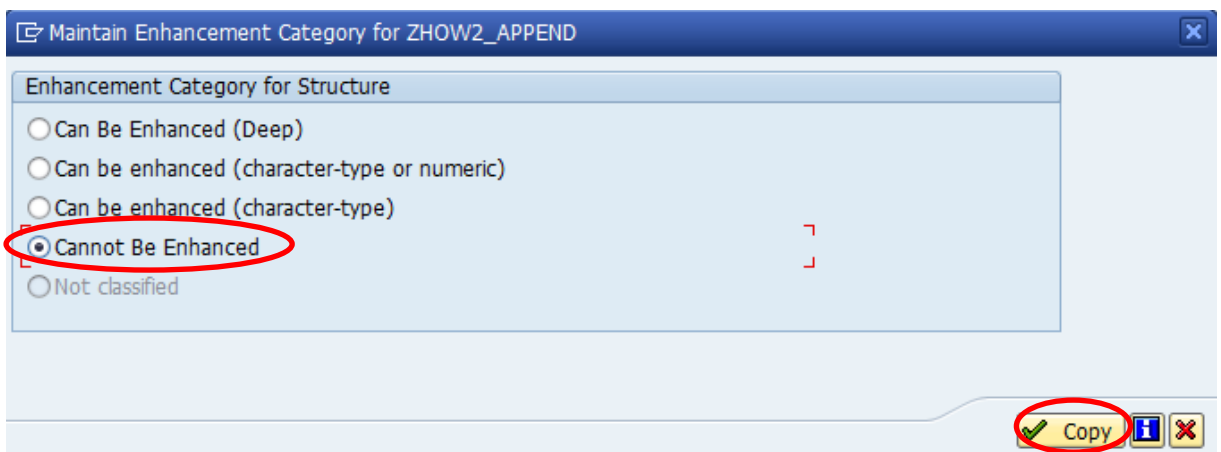
1.3. Enter append name and confirm with [Continue] button (or [Enter] key).



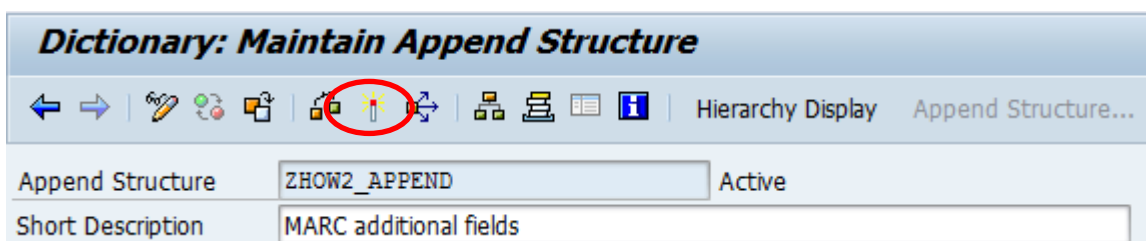
1.4. In Maintain Append Structure view simply complete [Short Description] and definition of our new field in [Components] tab as shown below.



1.5. From [Extras] menu choose [Enhancement Category ...] and set it to "Cannot Be Enhanced" (or whatever you need :) and confirm with [Copy] button.



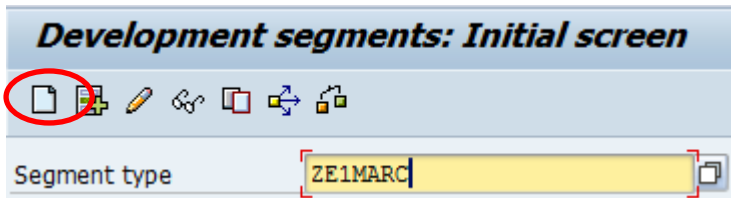
1.6. Activate your changes (and assign it to package and transport request if needed).



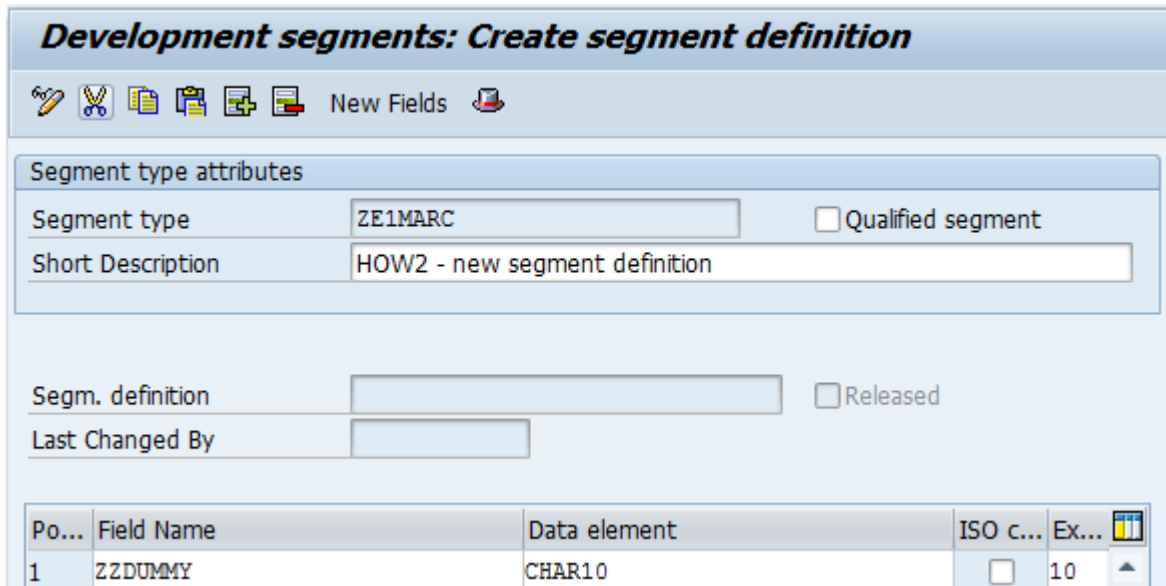
2. Enhance IDoc MATMAS03

Material Master mass maintenance transaction uses ALE inbound processing to save changes to materials. That is why we have to start with enhancing MATMAS03 IDoc with our ZZDUMMY append field.

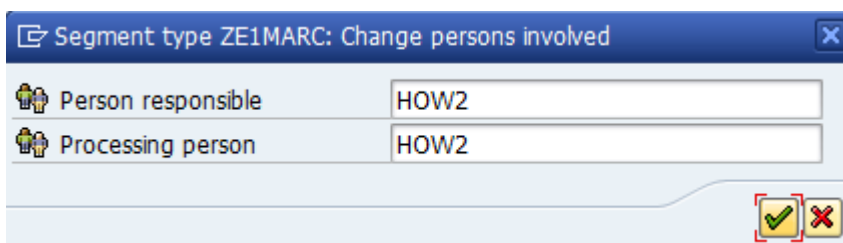
- 2.1. Start transaction WE31 and enter new segment name in [Segment type] field, then press [Create] button.



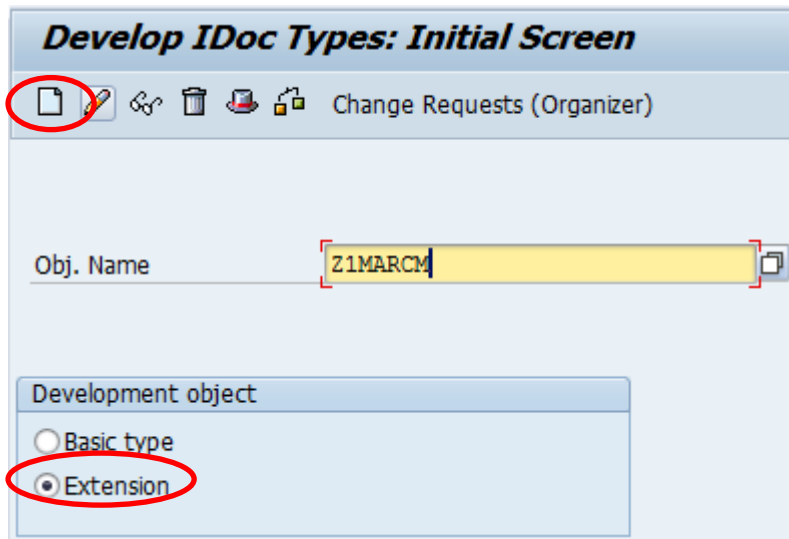
- 2.2. Complete [Short Description] and add appended field to the list as shown below. Save your changes.



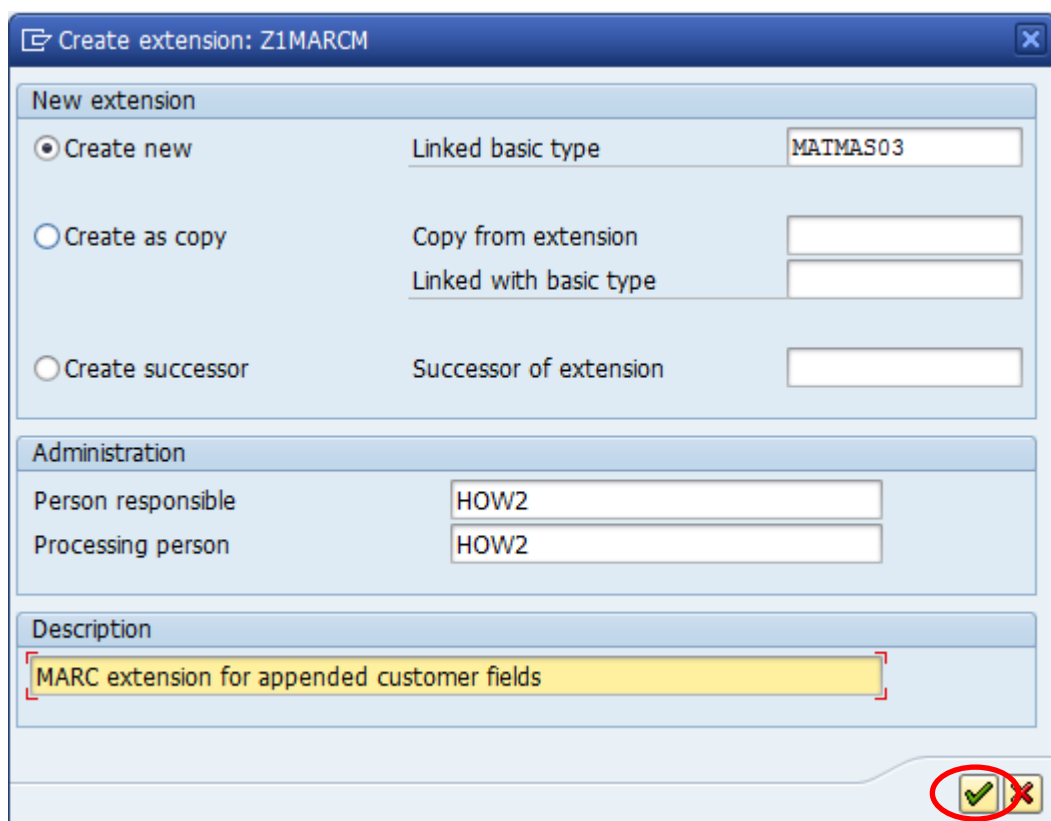
- 2.3. Confirm proposed values for "persons involved".



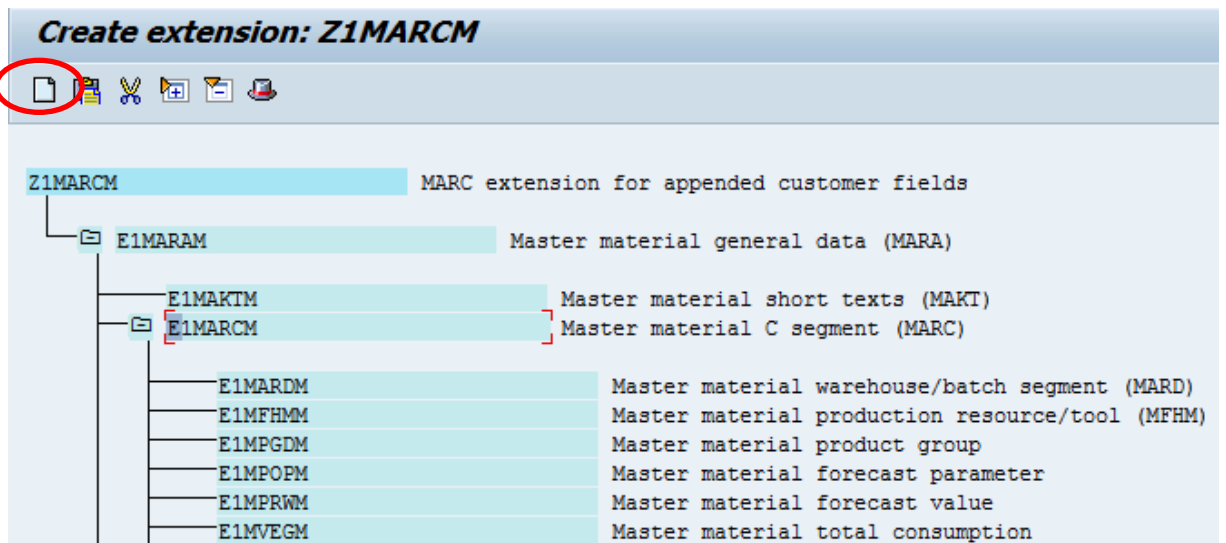
2.4. Start transaction WE30 to create new extension type for IDoc MATMAS03. Enter [Obj. Name], choose [Extension] option and click on [Create] button.



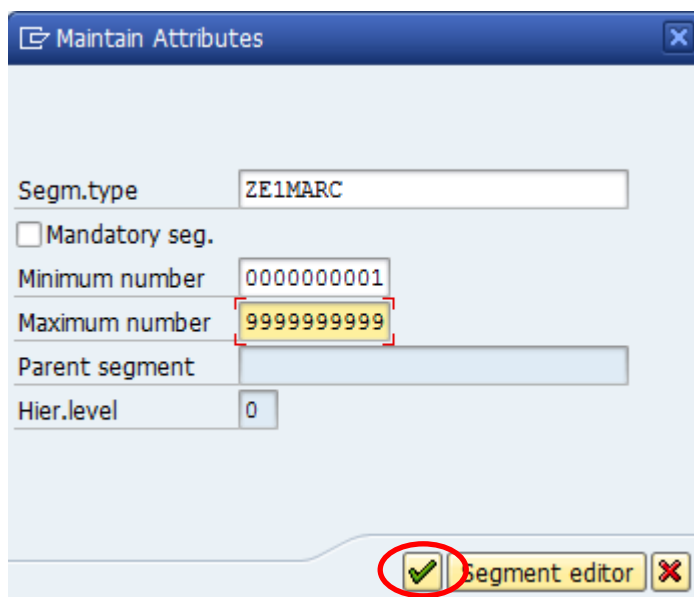
2.5. Enter MATMAS03 as [Linked basic type], provide description and confirm with [Continue] button.



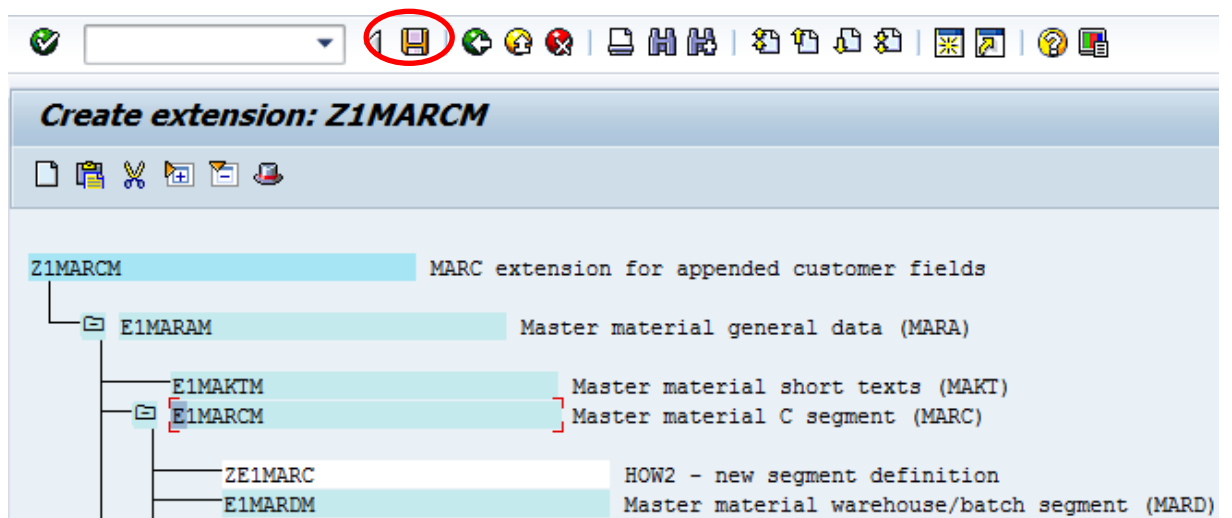
2.6. Mark E1MARCM segment by clicking it and then choose [Create segment] button.



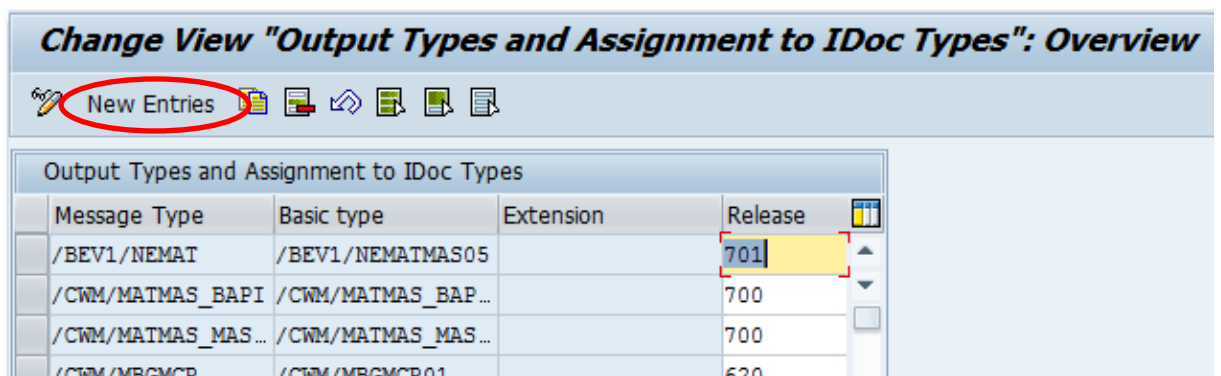
2.7. Now you have to maintain basic attributes of extension segment. Enter the name of segment type created in step 2.1. and maximum/minimum numbers as show below. Confirm with [Continue] button.



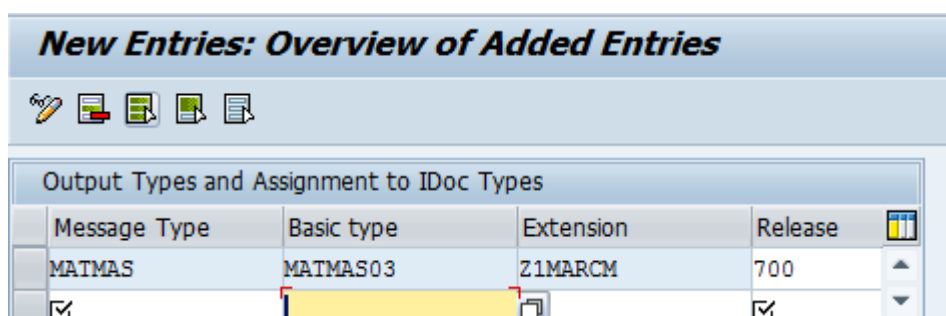
2.8. Save your changes



2.9. Now you have to assign newly created extension to message type and IDoc type. In order to do it you have to start transaction WE82, switch to "change mode" and click on [New Entries] button



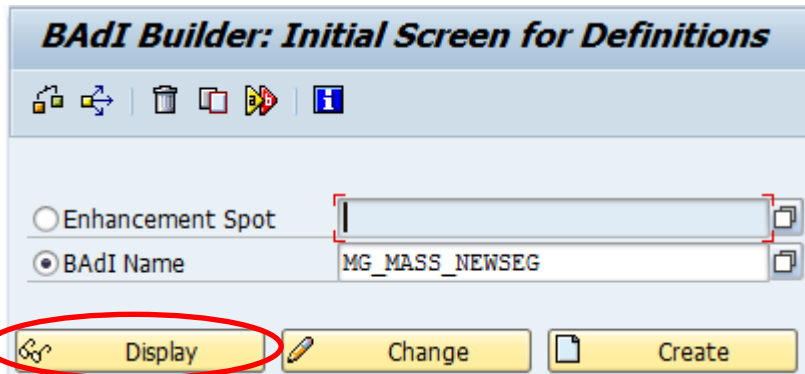
2.10. Enter your combination as shown below and save changes.



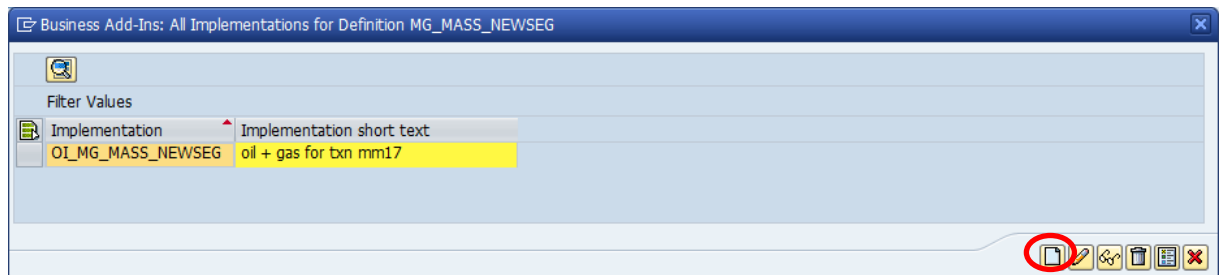
3. Create implementation for MG_MASS_NEWSEG BAdI

In order to populate the data into newly created segment you have to implement ADD_NEW_SEGMENT method of BAdI MG_MASS_NEWSEG. This is a tricky one so you can always refer to sample code (with comments) provided by SAP to have something to start with.

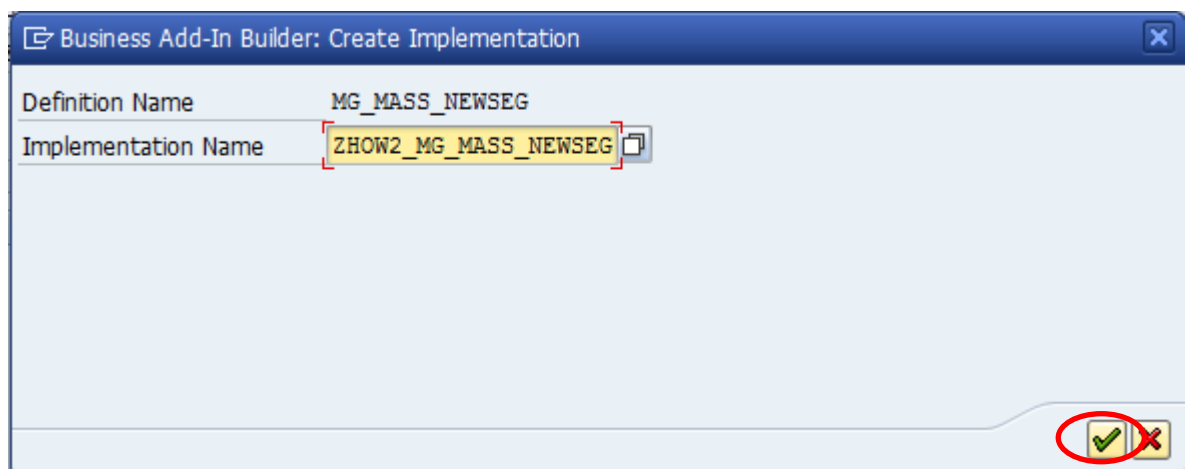
- 3.1. Start transaction SE18 and enter "MG_MASS_NEWSEG" into [BAdI Name] field then press [Display].



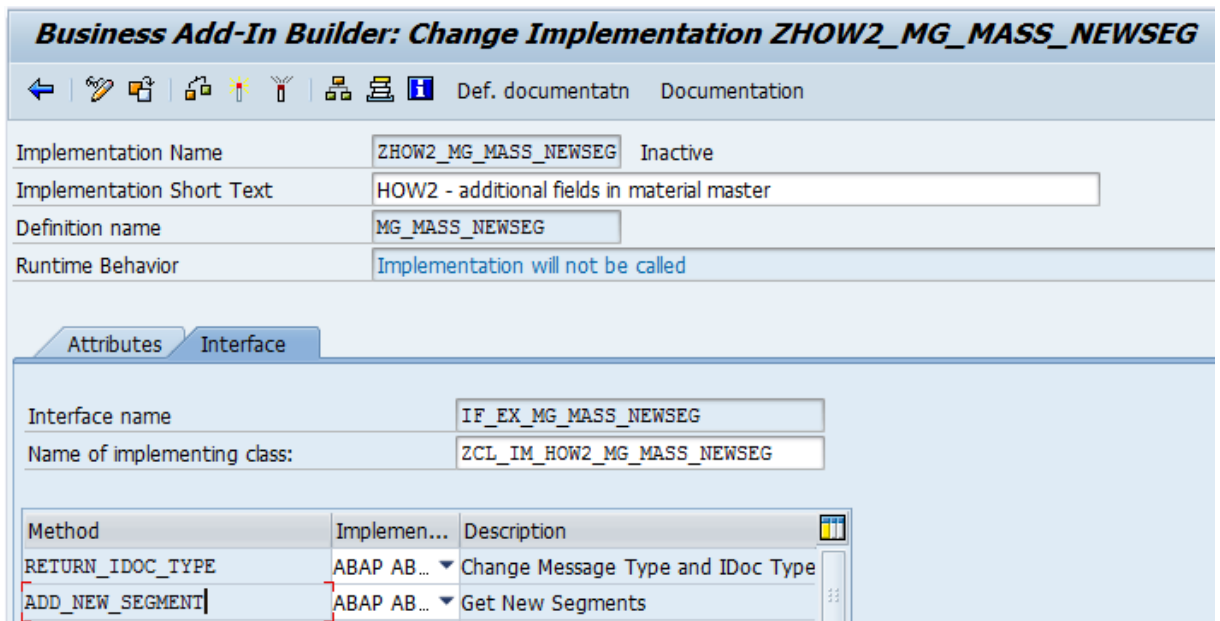
- 3.2. From menu [Implementation] choose [Overview]. Click on [Create] button.



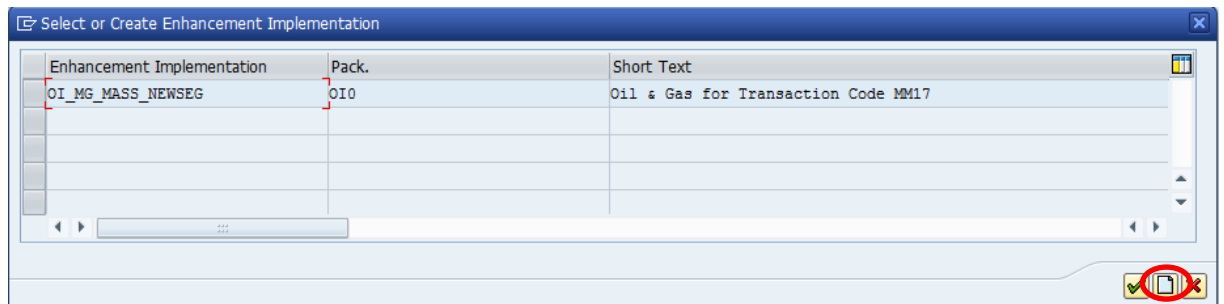
- 3.3. Fill [Implementation Name] and confirm with [Continue] button.



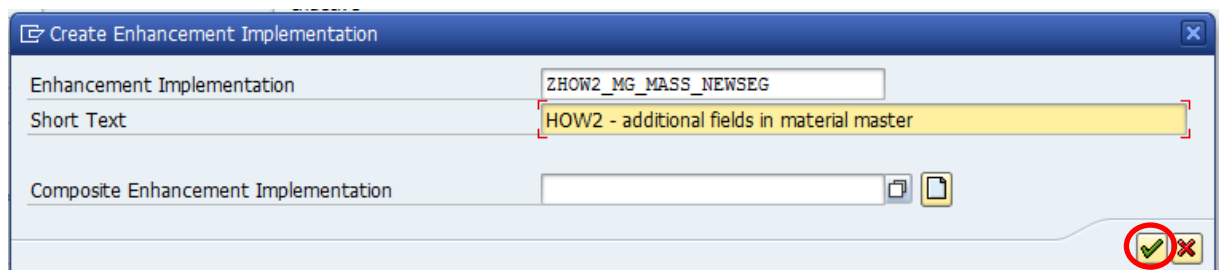
3.4. Enter [Implementation Short Text] and click on [Save].



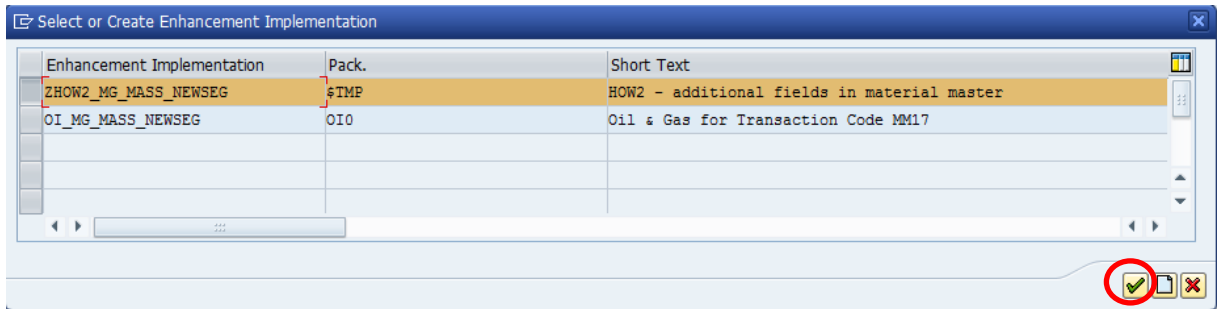
3.5. Because new BAdIs are part of enhancement framework in order to implement it you have to create new enhancement implementation. Click on [Create Enhancement Implementation] button to do it.



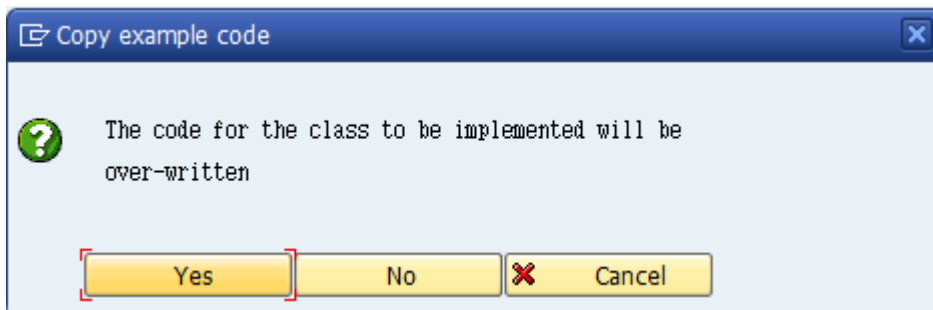
3.6. Complete information as shown below and confirm with [Creation of Enhancement] button.



3.7. Mark your newly created implementation and confirm your choice with [Select Specified Enhancement Implementation] button.



3.8. If you would like to start with SAP sample code choose from menu [Goto] option [Sample code]->[Copy] and confirm with [Yes] button.



3.9. Below you will find a sample code with basic logic (prepared only for test purposes).

3.9.1. Method: RETURN_IDOC_TYPE (remove the code for IS-OIL!)

```
METHOD if_ex_mg_mass_newseg~return_idoc_type .
ENDMETHOD.
```

3.9.2. Method: ADD_NEW_SEGMENT

```
METHOD if_ex_mg_mass_newseg~add_new_segment .
*-----
* HOW.2 (2013-12-12) Begin Types declaration of customer specific IDoc segments
*-----
  TYPES: BEGIN OF lty_zelmarcm,
          docnum   TYPE edidc-docnum,
          pointer  TYPE sy-tabix.
          INCLUDE  TYPE zelmarc AS data. "<<< Custom segment!
  TYPES: END OF lty_zelmarcm.
  TYPES: lty_zelmarcm_tab TYPE STANDARD TABLE OF lty_zelmarcm.
*-----
* HOW.2 (EndOF) Begin Types declaration of customer specific IDoc segments
*-----
*-----
* HOW.2 (2013-12-12) Local data declarations
*-----
  DATA: lv_tabix      TYPE sytabix,
         ls_idoc_data  TYPE edidd,
         lt_elmaram    TYPE mgmc_elmaram_tab,
         lt_elmarcm    TYPE mgmc_elmarcm_tab,
         ls_zelmarcm   TYPE lty_zelmarcm,
```

```

        lt_zelmarcm    TYPE lty_zelmarcm_tab.
FIELD-SYMBOLS: <ls_smarc>    TYPE marc,
               <ls_elmaram>  TYPE mgmc_elmaram,
               <ls_elmarcm>  TYPE mgmc_elmarcm.
*-----
* HOW.2 (EndOf) Local data declarations
*-----
*-----
* HOW.2 (2013-12-12) Initializations
*-----
* Copy the IDoc segment tables from BADI interface to local tables
  lt_elmaram = t_elmaram.
  lt_elmarcm = t_elmarcm.
*-----
* HOW.2 (EndOf) Initializations
*-----
*-----
* HOW.2 (2013-12-12) Logic
*-----
LOOP AT smarc ASSIGNING <ls_smarc>.

  CLEAR: ls_zelmarcm , lv_tabix.
  MOVE-CORRESPONDING <ls_smarc> TO ls_zelmarcm .

  READ TABLE lt_elmaram
    WITH KEY matnr = <ls_smarc>-matnr
    ASSIGNING <ls_elmaram>.
  IF sy-subrc NE 0.
    CONTINUE.
  ENDIF.

  READ TABLE lt_elmarcm
    WITH KEY docnum = <ls_elmaram>-docnum
           werks   = <ls_smarc>-werks
    ASSIGNING <ls_elmarcm>.
  IF sy-subrc NE 0.
    CONTINUE.
  ENDIF.

  CALL FUNCTION 'I_MASS_GET_INDEX'
    EXPORTING
      pointer = <ls_elmarcm>-pointer
    IMPORTING
      tabix   = lv_tabix.
  IF lv_tabix IS INITIAL.
    lv_tabix = 1.
  ELSE.
    lv_tabix = lv_tabix + 1.
  ENDIF.

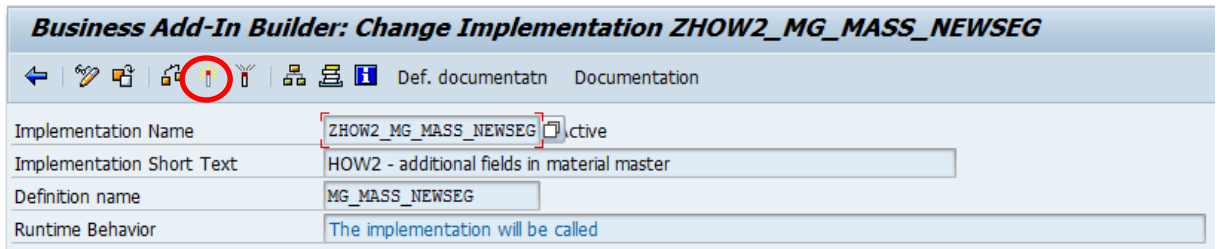
  ls_idoc_data-segnam = 'Z1MARCM' .           "<<< Customer segment name!
  ls_idoc_data-sdata  = ls_zelmarcm-data .
  ls_idoc_data-docnum = <ls_elmaram>-docnum.

  INSERT ls_idoc_data
    INTO t_idoc_data
    INDEX lv_tabix.

  ENDLOOP.
*-----
* HOW.2 (EndOf) Logic
*-----
ENDMETHOD.

```

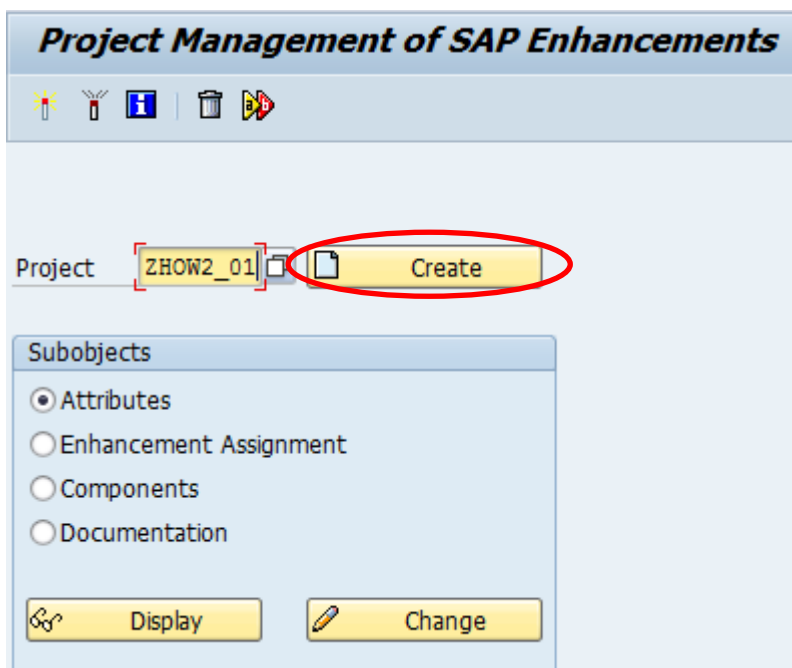
- 3.10. Exit code editor and activate BAdI implementation by clicking [Activate business add-in implementation] button. Ensure that [Runtime Behavior] shows “The implementation will be called” text.



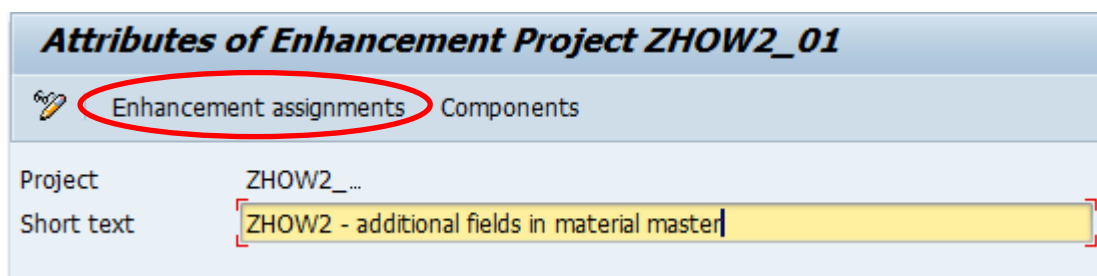
4. Implement customer exit MGV00001

In order to populate our ZZDUMMY field with data from IDoc during ALE processing you have to prepare implementation in function module EXIT_SAPLMV02_002 of customer exit MGV00001.

- 4.1. Start transaction CMOD, enter new project name in [Project] field and click on [Create] button.



- 4.2. Enter [Short text] and click on [Enhancement assignments] button.



- 4.3. Type in "MGV00001" in [Enhancement] field and click on [Components] button.



4.4. Double click on “EXIT_SAPLMV02_002” .

Change ZHOW2_01				
Enhancement assignments Enhancement				
Project				ZHOW2_01 ZHOW2 - additional fields in material master
Enhancement	Impl		Exp	MGV00001 Material Master (Industry): ALE Distribution
Function exit				EXIT_SAPLMV01_002 EXIT_SAPLMV02_002

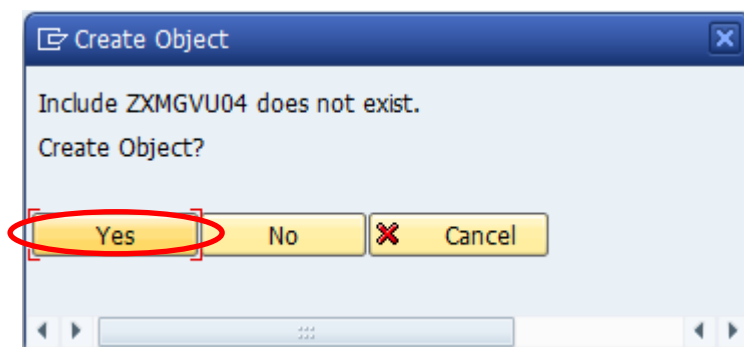
4.5. Double click on “ZXMGVU04”.

```

Function Builder: Display EXIT_SAPLMV02_002
Function module EXIT_SAPLMV02_002 Active
Attributes Import Export Changing Tables Exceptions
43:  **          APPLICATION_ERROR
44:  **          -----
45:
46:
47:  INCLUDE * ZXMGVU04 .
48:
49:
50:  ENDFUNCTION.
  
```

4.6. You will receive a status bar warning:

Program names ZX... are reserved for includes of exit function groups
 Press [Enter] button to skip it and confirm object creation by clicking [Yes] button.



4.7. Enter following code to populate MARC-ZZDUMMY field with data from IDoc.








```

*&-----*
*& Include          ZXMGVU04
*&-----*
*-----*
* HOW.2 (2013-12-12) Local data declarations
*-----*
DATA: ls_zelmarc TYPE zelmarc.
*-----*
* HOW.2 (EndOf) Local data declarations
*-----*
*-----*
* HOW.2 (2013-12-12) Logic
*-----*
IF f_cust_segment-segnam = 'Z1MARCM'.
  ls_zelmarc = f_cust_segment-sdata.
  IF ls_zelmarc-zzdummy = c_nodata.
    CLEAR ls_zelmarc-zzdummy.
  ELSE.
    IF ls_zelmarc-zzdummy IS INITIAL.
      res_fields-feldname = 'MARC-ZZDUMMY'.
      APPEND res_fields.
    ENDIF.
  ENDIF.
  f_marc_ueb-zzdummy = ls_zelmarc-zzdummy.
ENDIF.
*-----*
* HOW.2 (EndOf) Logic
*-----*

```

4.8. Go back to CMOD enhancement screen and activate EXIT_SAPLMV02_002 by clicking [Activate] button.

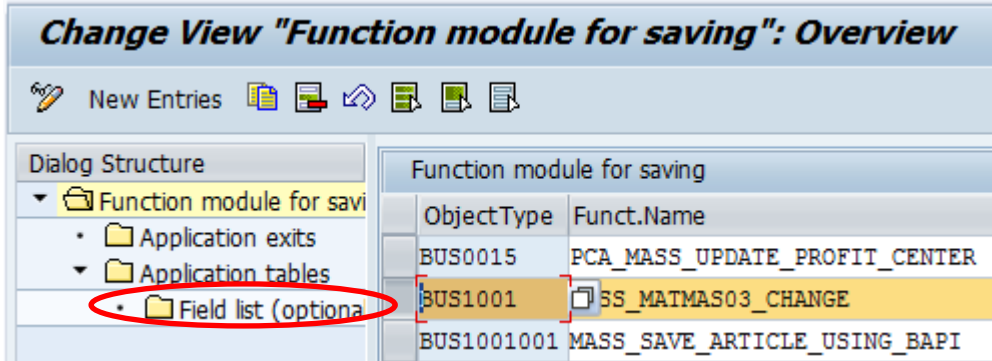
Change ZHOW2_01







 Enhancement assignments  Enhancement

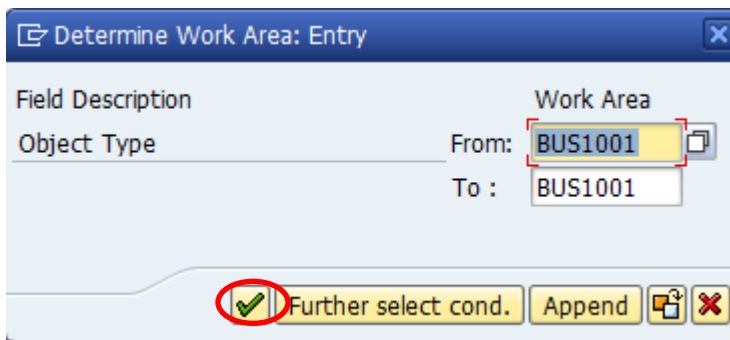
Project		✖		ZHOW2_01 ZHOW2 - additional fields in material master
Enhancement	Impl	✖	Exp	MGV00001 Material Master (Industry): ALE Distribution
Function exit	✔			EXIT_SAPLMV01_002 EXIT_SAPLMV02_002

5. MASS object customizing.

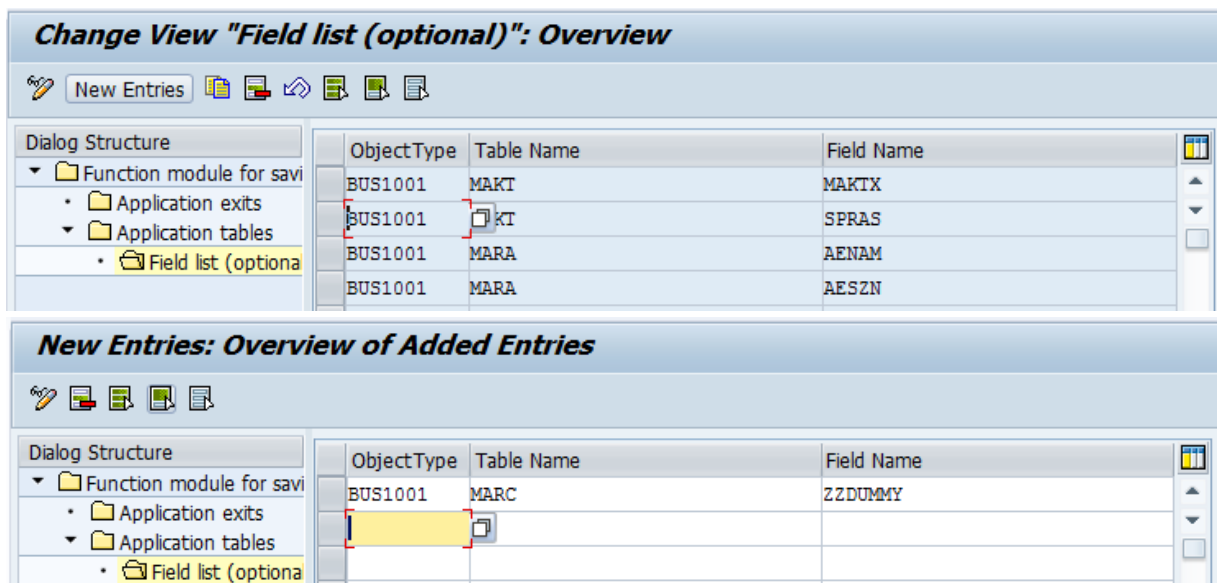
5.1. Start transaction MASSOBJ and mark BUS1001 row. Then double click on [Field list (optional)] folder.



5.2. Confirm "Work Area" by clicking [Continue] button.



5.3. Click [New Entries] button and enter BUS1001 – MARC – ZZDUMMY combination for your appended field. Save changes.



6. Field selection group settings.

6.1. Start OMSR transaction and click [New Entries] button.

Change View "Field Groups": Overview

New Entries

Field name in full	Short Description	Sel. gr...
*TC25T-TXT		66
*TCA55-BZTXT		67
*TCF13-FGRITX		66

6.2. Complete customizing as shown below and save it ("E" stands for "Purchasing", use F1 help for full list of available statuses).

Change View "Field Groups": Details

New Entries

Field name:

Character Field Length = 10

Field attributes (industry and retail)

Propose field cont.

Maint. status:

ALE field group:

Field attributes (retail only)

Restrict matl cat.:

Copy field content

Incl. initial values

6.3. Assign MARC-ZZDUMMY field to [Selection group] (the best option would be to choose selection group which already exists and is assigned to a field with similar properties) and save.

Change View "Field Groups": Overview

New Entries

Field name in full	Short Description	Sel. gr...	
MARC-WZEIT	Total replenishment lead time (in workdays)	57	
MARC-XCHPF	Batch management requirement indicator	75	
MARC-XMCNG	Negative stocks allowed in plant	83	
MARC-ZZDUMMY	Character Field Length = 10	83	

7. Test your code.

That is all you needed to prepare. Now you should be able to change value of ZZDUMMY field from transaction MM17.