

Profit center determination via substitution rule in ECC

Steps to fulfill requirement

1. CRM: Sales group is part of BAPISDCOSV and is passed to ECC
2. ECC : Extend BAPISDCOSV with sales data (i.e. sales group)
3. ECC: use exit IAOM_ATTRIBUTES to add sales group to characteristic table (ls_obj_desc-attributes.)
4. ECC: User exit **PERFORM** CALL_USER_EXIT_001 (RKECRM_PCA_40) to populate structure LD_PCASUB , field VKGRP with sales group used by substitution rule

CRM

Standard logic put the (ECC) sales group into the CO interface.

ECC

Extend Structure BAPISDCOSV with Sales Data

The sales data fields that do exist in the parameter structure BAPISDCOSV are missing in ECC

CRM:

Dictionary: Display Structure

Structure: BAPISDCOSV Active
Short Description: CRM Sales: Servicefelder zur CO-Kontierungsfindung

Attributes Components Entry help/check Currency/quantity fields

Predefined Type 45 / 51

Component	Typing Method	Component Type	Data Type	Length	Deci...	Short Description
PRODUCT_ID	1 Types	COMT_PRODUCT_ID	CHAR	40		0 Product ID
AC_NOCOST	1 Types	CRMT_NOCOST	CHAR	1		0 No Costs Expected
SALESORG	1 Types	VKORG	CHAR	4		0 Sales Organization
DISTR_CHAN	1 Types	VTWEG	CHAR	2		0 Distribution Channel
DIVISION	1 Types	SPART	CHAR	2		0 Division
SALES_GRP	1 Types	VKGRP	CHAR	3		0 Sales group
SALES_OFF	1 Types	VKBUR	CHAR	4		0 Sales office

ECC

Dictionary: Display Structure

Structure: BAPISDC0SV Active
Short Description: CRM Sales: Service Fields for CO Account Assignment Determ.

Attributes Components Entry help/check Currency/quantity fields

Predefined Type 44 / 52

Component	Typing Method	Component Type	Data Type	Length	Deci...	Short Description
TOP_PROCESS_TYPE	1 Types	CRM TOP_PROCESS_TYP...	CHAR	4		0 Leading Business Transaction Type
PRODUCT_ID	1 Types	CONT_PRODUCT_ID_CO	CHAR	40		0 Product ID
AC_NOCOST	1 Types	TP_COSTS	CHAR	1		0 No Expected Costs
APPEND	1 Types	ZBAPI_SALES_DATA		0		0 Sales Data
SALESORG	1 Types	VKORG	CHAR	4		0 Sales Organization
DISTR_CHAN	1 Types	VTWEG	CHAR	2		0 Distribution Channel
DIVISION	1 Types	SPART	CHAR	2		0 Division
SALES_GRP	1 Types	VKGRP	CHAR	3		0 Sales Group
SALES_OFF	1 Types	VKBUR	CHAR	4		0 Sales Office

ECC: use exit to add sales group to characteristic table

Add new filter value via TC IAOM4

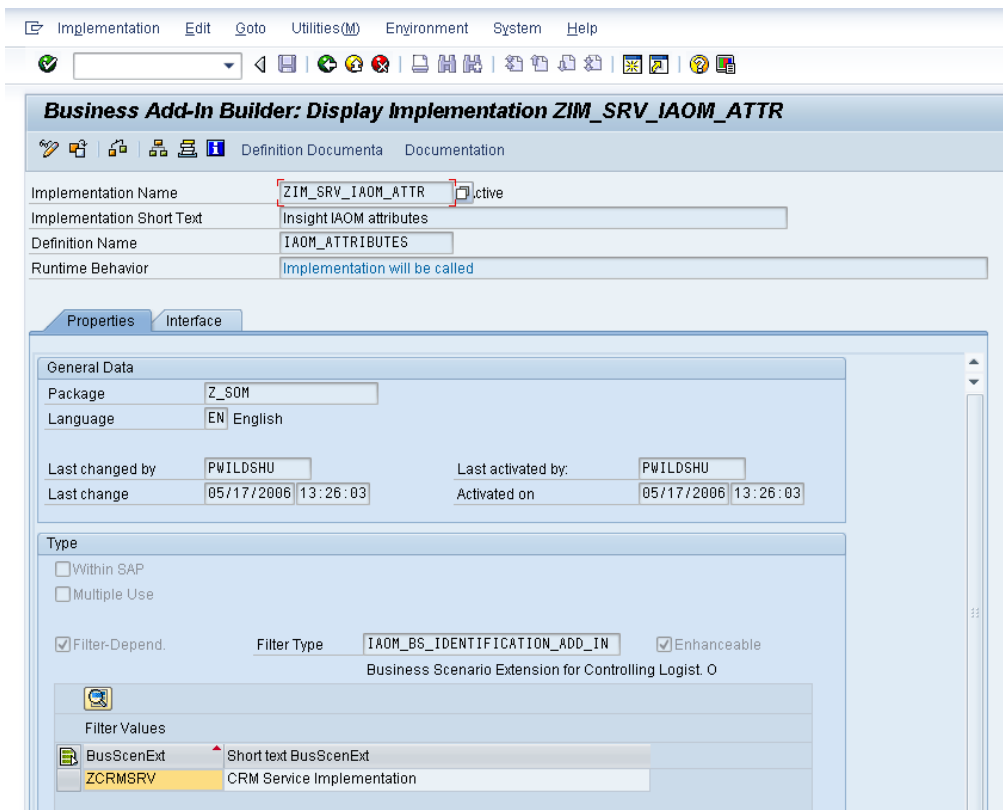
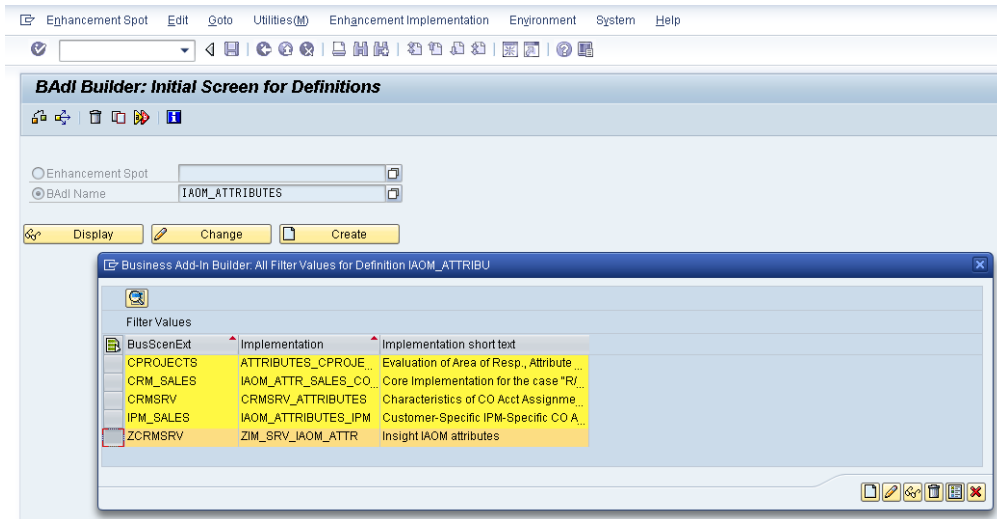
Table View Edit Goto Selection Utilities(M) System Help

Display View "Business Scenario Extensions Control Table": Overview

Business Scenari...	Business Scenario Extens.	Business Scenario	Sort
CPROJECTS	cProjects Implementation	CPROJECTS	1
CRMFIN	CRM Financing Implementation	CRMFIN	1
CRMSRV	CRM Service Implementation	CRMSRV	1
CRM_SALES	CRM Sales Implementation	CRM_SALES	1
GRANTOR	Grantor Management	GRANTOR	1
IPM_SALES	IPM Contracts	CRM_SALES	10
OM	Outsourced Manufacturing	OM	1
ZCRMSRV	CRM Service Implementation	CRMSRV	2

Add Sales Group and Responsible Cost Center

Create new implementation for BADI IAOM_ATTRIBUTES



```

* METHOD if_ex_iaom_attributes~change_attrbts
* -----*
* Change next internal order attributes:
* 1. Sales Group
* 2. Responsible Costcenter
* -----*
* Pieter Wildschut 01/11/2006 initial version
* -----*
METHOD if_ex_iaom_attributes~change_attrbts.

    DATA: dfies_tab          TYPE TABLE OF dfies ,
           dfies             TYPE dfies,
           tabname           TYPE dfies-tabname.
    DATA: lv_funcname        TYPE rs38l_fnam.
    DATA: lv_object_no       TYPE j_objnr.
    DATA: ls_bapisdcosv      TYPE bapisdcosv.

    FIELD-SYMBOLS <process_type> TYPE ANY.
    FIELD-SYMBOLS <object_id> TYPE ANY.
    FIELD-SYMBOLS <current_object_id> TYPE ANY.

    ASSIGN COMPONENT 'PROCESS_TYPE' OF STRUCTURE i_data
    TO <process_type>.

    ASSIGN COMPONENT 'OBJECT_ID' OF STRUCTURE i_data
    TO <object_id>.

    ASSIGN COMPONENT 'CURRENT_OBJECT_ID' OF STRUCTURE i_data
    TO <current_object_id>.

    CONCATENATE 'OR' <object_id> INTO lv_object_no.

    ls_bapisdcosv = i_data.
* get current attributes and merge them with new
CALL FUNCTION 'ZSM_GET_CURR_ATTR'
    EXPORTING
        iv_object_id    = <current_object_id>
    IMPORTING
        es_bapisdcosv   = ls_bapisdcosv
    EXCEPTIONS
        incorrect_call  = 1
        error           = 2
        not_assigned    = 3
        OTHERS          = 4.

    IF sy-subrc <> 0 AND sy-subrc NE 3.
        EXIT.
    ENDIF.

*
    FIELD-SYMBOLS <lw_bapisdcosv> TYPE ANY.
    DATA: lw_bapisdcosv          TYPE REF TO data.
    CREATE DATA lw_bapisdcosv TYPE bapisdcosv.
    ASSIGN lw_bapisdcosv->* TO <lw_bapisdcosv>.

    FIELD-SYMBOLS <lp_bapisdcosv> TYPE ANY.
    DATA: lp_bapisdcosv          TYPE REF TO data.
    CREATE DATA lp_bapisdcosv TYPE bapisdcosv.
    ASSIGN lp_bapisdcosv->* TO <lp_bapisdcosv>.

* sales group
CALL FUNCTION 'ZSM_FM_INTORD_ATTR_VKGRP'

```

```

EXPORTING
    i_data                = i_data
    i_obj_stndrd_acc_attributes = i_obj_stndrd_acc_attributes
CHANGING
    iaomt_message_tab      = e_messages
    et_object_attributes   = c_object_attributes.

* responsible costcenter
CALL FUNCTION 'ZSM_FM_INTORD_ATTR_KOSTV'
EXPORTING
    i_data                = i_data
    i_obj_stndrd_acc_attributes = i_obj_stndrd_acc_attributes
CHANGING
    iaomt_message_tab      = e_messages
    et_object_attributes   = c_object_attributes.

ENDMETHOD.

FUNCTION ZSM_FM_INTORD_ATTR_VKGRP.
*-----
*""Local Interface:
* IMPORTING
*     REFERENCE(I_DATA) TYPE ANY
*     REFERENCE(I_OBJ_STNDRD_ACC_ATTRIBUTES) TYPE
*                                     IAOMT_OBJECT_ATTRIBUTE_TAB
* CHANGING
*     REFERENCE(IAOMT_MESSAGE_TAB) TYPE IAOMT_MESSAGE_TAB
*     REFERENCE(ET_OBJECT_ATTRIBUTES) TYPE
*                                     IAOMT_OBJECT_ATTRIBUTE_TAB
*-----

    lv_data = i_data.

* Add attribute
CALL FUNCTION 'ZSM_ASSIGN_VALUE'
EXPORTING
    iv_fieldname          = 'SALES_GRP'
    iv_data               = lv_data
IMPORTING
    et_object_attributes = et_object_attributes
EXCEPTIONS
    invalid_fieldname    = 1
    error                = 2
    OTHERS               = 3.
IF sy-subrc NE 0.
ENDIF.

ENDFUNCTION.

FUNCTION zsm_assign_value.
*-----
*""Local Interface:
* IMPORTING
*     REFERENCE(IV_FIELDNAME) TYPE FIELDNAME
*     REFERENCE(IV_DATA) TYPE BAPISDCOSV

```

```

* " EXPORTING
* " REFERENCE(ET_OBJECT_ATTRIBUTES) TYPE IAOMT_OBJECT_ATTRIBUTE_TAB
* " EXCEPTIONS
* " INVALID_FIELDNAME
* " ERROR
* "-----

FIELD-SYMBOLS : <f> TYPE any,
                <val> TYPE any.

DATA: lv_object_attributes TYPE iaom_object_attribute.
CLEAR lv_object_attributes.

READ TABLE et_object_attributes INTO lv_object_attributes
WITH KEY data_element = iv_fieldname.

IF sy-subrc NE 0 OR lv_object_attributes-value IS INITIAL.

    ASSIGN COMPONENT iv_fieldname OF STRUCTURE iv_data TO <f>.
    IF sy-subrc NE 0.
        RAISE invalid_fieldname.
    ENDIF.

    ASSIGN COMPONENT 'VALUE' OF STRUCTURE lv_object_attributes TO <val>.
    IF sy-subrc NE 0.
        RAISE error.
    ENDIF.

    <val> = <f>.

* add attribute
    lv_object_attributes-data_element = iv_fieldname.
    APPEND lv_object_attributes TO et_object_attributes.
ENDIF.

ENDFUNCTION.

```

Activate Customer Function 001 for enhancement PCACRM01

Code: Populate the sales group into parameter CH_PCASUB so the substitution rule can use it.

SMOD:

Components in SAP Enhancement PCACRM01

Function module exits

Function module	Short Text
EXIT_RKECRM_PCA_40_001	Substitution Logic for Derivation of Profit Center fr ...

Function codes

Program	Code	Short text

```
FUNCTION EXIT_RKECRM_PCA_40_001.
* "-----
* " "Lokale Schnittstelle:
* "      IMPORTING
* "          VALUE(IM_DATE) LIKE COBL-BUDAT
* "          VALUE(IM_KOKRS) LIKE COBL-KOKRS
* "          VALUE(IM_BUKRS) LIKE COBL-BUKRS
* "          VALUE(IM_WERKS) LIKE MARC-WERKS
* "          VALUE(IM_MATNR) LIKE MARC-MATNR
* "          VALUE(IM_BUKRS_OF_WERKS) LIKE COBL-BUKRS
* "          VALUE(IM_PRCTR_IN_CHARTAB) TYPE C
* "      TABLES
* "          T_CHARTAB STRUCTURE IAOM_OBJECT_ATTRIBUTE
* "      CHANGING
* "          REFERENCE(CH_PCASUB) LIKE PCASUB STRUCTURE PCASUB
* "          REFERENCE(CH_PRCTR) LIKE COBL-PRCTR
* "          REFERENCE(CH_CALL_SUBSTITUTION) TYPE C
* "          REFERENCE(CH_SUBSTID) LIKE GB92-SUBSTID
* "-----
```

```
INCLUDE ZXKECRM40PCAU01 .
```

```
ENDFUNCTION.
```

```
*&-----*
*& Include          ZXKECRM40PCAU01
*&-----*
```

```
DATA: LS_CHARTAB TYPE IAOM_OBJECT_ATTRIBUTE.
```

```
LOOP AT T_CHARTAB INTO LS_CHARTAB WHERE DATA_ELEMENT = 'SALES_GRP'.
```

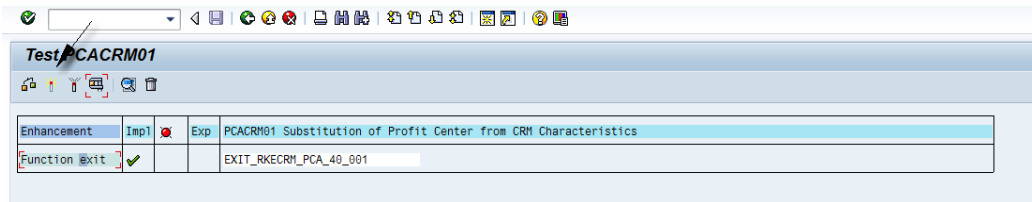
```
ENDLOOP.
```

```
IF SY-SUBRC = 0.
```

```
CH_PCASUB-VKGRP = LS_CHARTAB-VALUE.
```

```
ENDIF.
```

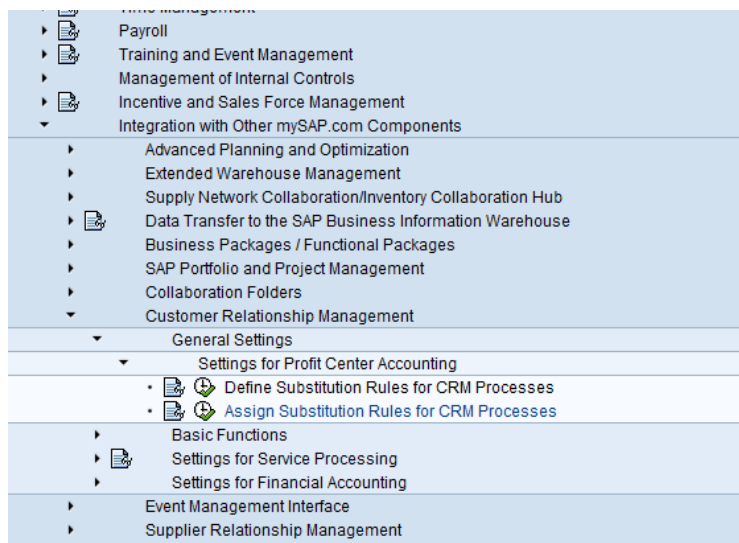
Do not forget to activate!



The screenshot shows the SAP transaction PCACRM01. The title bar reads "Test PCACRM01". Below the title bar is a toolbar with various icons. The main area contains a table with the following data:

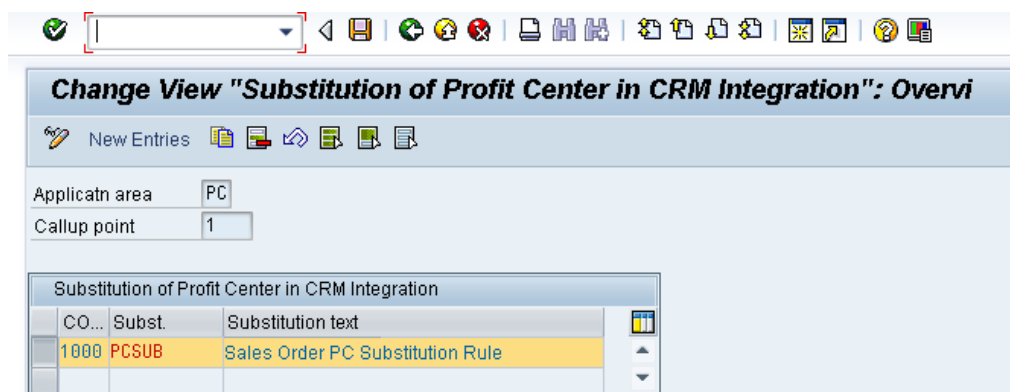
Enhancement	Impl	Exp	PCACRM01 Substitution of Profit Center from CRM Characteristics
Function Exit	✓		EXIT_RKCRM_PCA_40_001

Profit Center substitution rule:



The screenshot shows the SAP menu path for setting up a Profit Center substitution rule. The menu is structured as follows:

- Integration with Other mySAP.com Components
 - Advanced Planning and Optimization
 - Extended Warehouse Management
 - Supply Network Collaboration/Inventory Collaboration Hub
 - Data Transfer to the SAP Business Information Warehouse
 - Business Packages / Functional Packages
 - SAP Portfolio and Project Management
 - Collaboration Folders
 - Customer Relationship Management
- General Settings
 - Settings for Profit Center Accounting
 - Define Substitution Rules for CRM Processes
 - Assign Substitution Rules for CRM Processes
- Basic Functions
 - Settings for Service Processing
 - Settings for Financial Accounting
- Event Management Interface
- Supplier Relationship Management



The screenshot shows the SAP transaction "Change View 'Substitution of Profit Center in CRM Integration': Overview". The title bar reads "Change View 'Substitution of Profit Center in CRM Integration': Overview". Below the title bar is a toolbar with various icons. The main area contains the following fields and table:

Applicatn area: PC

Callup point: 1

CO...	Subst.	Substitution text
1000	PCSUB	Sales Order PC Substitution Rule

Function PI_CRM_GET_PROFITCENTER

Program RKECRM_PCA_40

```
...
IF LD_RC EQ CO_NOthing_FOUND.
    LD_PRCTR_IN_CHARTAB = CO_FALSE.
ELSE.
    LD_PRCTR_IN_CHARTAB = CO_TRUE.
ENDIF.

IF ( NOT LD_MATNR IS INITIAL ) AND
   ( NOT LD_WERKS IS INITIAL ) AND
   ( LD_PRCTR_IN_CHARTAB EQ CO_FALSE ) AND "do not change given value
   ( LD_BUKRS EQ LD_BUKRS_OF_WERKS ). "only non-X-company
PERFORM GET_PRCTR_FROM_MARC USING LD_MATNR
                                   LD_WERKS
                                   LD_KOKRS
                                   CHANGING LD_PRCTR.
ENDIF.

IF LD_PRCTR IS INITIAL.
    PERFORM GET_DEFAULT_PRCTR USING LD_KOKRS "to make this info
                                           "available in the
                                           CHANGING LD_PRCTR. "user exit
ENDIF.

PERFORM GET_SUBSTID USING LD_KOKRS
                       CHANGING LD_SUBSTID.

IF LD_SUBSTID IS INITIAL OR
   LD_PRCTR_IN_CHARTAB EQ CO_TRUE.
    LD_CALL_SUBSTITUTION = CO_FALSE.
ELSE.
    LD_CALL_SUBSTITUTION = CO_TRUE.
ENDIF.

* call user exit

PERFORM CALL_USER_EXIT_001
      USING PT_CHARTAB
            LD_KOKRS
            LD_BUKRS
            LD_WERKS
            LD_MATNR
            LD_BUKRS_OF_WERKS
            LD_PRCTR_IN_CHARTAB
            P_DATE
      CHANGING LD_PCASUB
              LD_PRCTR
              LD_CALL_SUBSTITUTION
              LD_SUBSTID.

IF LD_CALL_SUBSTITUTION EQ CO_TRUE.
* call substitution
  CLEAR LD_PCASUB-PRCTR.
```

```

        PERFORM CALL_SUBSTITUTION USING LD_SUBSTID
                                CHANGING LD_PCASUB.
        IF NOT LD_PCASUB-PRCTR IS INITIAL.
            LD_PRCTR = LD_PCASUB-PRCTR.
        ENDIF.
    ENDIF.

* PRCTR_NOT_FOUND exception -> derive PRCTR in CO-PA
    PERFORM CHECK_FOR_INITIAL_PRCTR USING LD_PRCTR
                                CHANGING P_RC
                                P_RETURN.

    IF p_rc NE co_success. EXIT. ENDIF.

* check if pc is valid
    PERFORM CHECK_PRCTR USING LD_PRCTR
                                LD_KOKRS
                                LD_BUKRS
                                P_DATE
                                CHANGING P_RC
                                P_RETURN.

    IF P_RC EQ CO_SUCCESS.
        P_PRCTR = LD_PRCTR.
        P_RC = CO_SUCCESS.
        EXIT.
    ELSE.
        PERFORM GET_DEFAULT_PRCTR USING LD_KOKRS
                                CHANGING LD_PRCTR.

        IF NOT LD_PRCTR IS INITIAL.            "dummy PC can be used
            P_PRCTR = LD_PRCTR.
            P_RC = CO_SUCCESS.
            EXIT.
        ENDIF.

* PRCTR_NOT_VALID exception -> derive PRCTR in CO-PA
* p_rc and p_return have been set within FORM check_prctr
    CLEAR P_PRCTR.
    EXIT.
    ENDIF.

ENDFORM.

```