Customizing Characteristic Relationships in BW-BPS with Function Modules

Applies to:

BW-BPS (Ver. 3.5 and BI 7.0)

SEM-BPS (Ver 3.2 onwards)

Summary

This paper discusses the definition of a 'exit' type characteristic relationship in BW-BPS using ABAP function modules and its functional application through a practical example.

Author: Aditya Naik

Created on: 23 August 2006

Author Bio

Aditya Naik is a BI Analyst having worked on BW and BPS projects. He is a certified Business Intelligence Solution Consultant (NetWeaver '04) and has over 4 years of experience in the SAP – BI space. He has worked as a BW and BPS Analyst on various assignments as well as a faculty at SAP Labs.

Table of Contents

Introduction

Characteristic Relationships are used in BPS in order to define dependencies between the values of the masterdata that characterize the business scenario. In many cases, these dependencies are a bit more complicated than can be easily defined and are the result of programmed logic. The ***Repost*** function is the only function that allows you to change the values of the characteristics and is thus a pretty powerful function.

Utilizing the power of ABAP to determine the characteristic relationships, and the Repost function to assign the values, we can automate the assignment of characteristic values in a planning area.

In this paper, we shall take a look at the definition of characteristic relationships through ABAP as well as glance through the two important functions that utilize characteristic relationships.

Business Scenario

The *Profit Center* is determined on the basis of the *CSR* (Customer Sales Representative) and the *Material*. This reference data is loaded into an ODS in BW from the source system. Planning is however not done at the profit center level. In other words, the planner does not enter the *Profit Center*; it needs to be determined automatically by the system while saving the data in a BPS Planning Layout.

Solution Overview

We define a characteristic relationship on the basis of the *Material* and the *CSR* fields, to derive the *Profit Center*, reading data from the ODS. We then utilize the planning function '***Repost (Characteristic Relationship)***' in order to utilize the defined char. relationship and populate the *Profit Center* field in the planning data.

This planning function is then utilized in a planning sequence in addition to other functions. The planning sequence is fired off before the data is saved so as to calculate the values for the characteristics in the transaction data.

Background

Characteristic Relationships

Characteristic relationships are built in order to restrict the combinations occurring through the transaction data to a valid set as defined by the business rules.

For example: Consider *Material* and *Material group* are two characteristics. *Material Group* is defined as an attribute of Material in BW. The Masterdata looks like the table below:

Material	Material Group
M0001	GRP1
M0002	GRP1
M0003	GRP1
M0004	GRP2
M0005	GRP2

While defining manual planning layouts, there could be situations, where you would want to display all the 'possible' combinations even though they do not contain any transaction data. In this case, characteristic relationships help to restrict the combinations to only those that are functionally possible, providing a correct business perspective. Thus, in case we were using these two characteristics in a manual planning layout with the option '*All Possible Characteristic Combinations*', Fig 1 shows the layout as it would appear with characteristic relationships defined for proposal and Fig 2 shows the layout without the characteristic relationships.

Fig 1

Material	Material Group	Key Fig 1	Key Fig 2	Key Fig 3
M0001	GRP1			
M0002	GRP1			
M0003	GRP1			
M0004	GRP2			
M0005	GRP2			

Fig 2

Material	Material Group	Key Fig 1	Key Fig 2	Key Fig 3
M0001	GRP1			
M0002	GRP1			

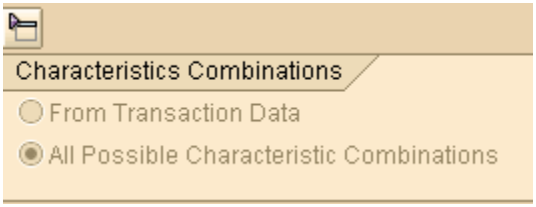
M0003	GRP1			
M0004	GRP1			
M0005	GRP1			
M0001	GRP2			
M0002	GRP2			
M0003	GRP2			
M0004	GRP2			
M0005	GRP2			

If, there are no characteristic relationships defined, the combination of the two characteristics could result in transaction data having 10 different combinations instead of the 5 mentioned above (each material with each material group). This would result in erroneous transaction data.

In this case, we define a characteristic relationship of the type '*attribute*'. The other types possible are *Reference Data* and *Hierarchy*. When the relationship defined cannot be represented by these, we can use the type '*Exit*' to determine the same through function modules.

Derivation, Proposal and Combination Check.

The restrictions defined in the characteristics relationships are used in three modes: Derivation, Proposal and Combination Check. In Derivation, values are determined for those target characteristics using the values of the source characteristics. Proposals are used to calculate the valid combinations while populating the manual planning layouts when the option 'All Possible Characteristic Combinations' is chosen.



Combination Check is used to check whether the existing combinations that have either been generated through the planning function or through user entries are correct.

Characteristic Relationships are thus used mainly

- To generate correct characteristic combinations using planning function 'REPOST (Characteristic Relationships)'
- To generate valid combinations of characteristic data in manual planning layouts
- Delete invalid combinations that exist in the transaction data using the planning function 'DELETE (Invalid Combinations)'.

Reposting Data

The planning function ***REPOST*** allows you to change the values of the characteristics in the planning package. In all the other functions, you're allowed to change the values of the key figures and this is where the Repost function is different from others. Repost has two variants: ***Repost*** and ***Repost (Characteristic Relationships)***.

You can choose which fields are to be changed and also select conditions under which they can be changed. Functionally, both are similar and both alter the values of the characteristics in the planning package. However, there are differences between the two in terms of the restrictions on the usage of the '*Repost (Characteristic Relationships)*' function.

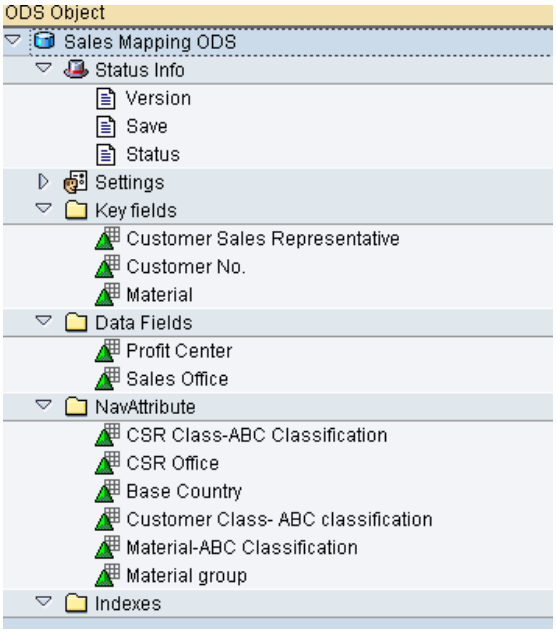
Firstly, you can only create a function of the type *Repost (Characteristic Relationships)* in a planning level that contains all the characteristics and key figures. With the Repost function, you can also restrict which key figures the function works on by choosing the field '*Key figure name'* in the Selections. This is not possible in the *Repost (Characteristic Relationships)* function. Also, only those fields which are derived in the characteristic relationships in the planning area, are available in the field list for Repost (Characteristic Relationships). You can also check the details of the derived fields using a pushbutton provided

In the Parameter group for Repost, you define the selections, and the actual values to be reposted in the function. However, the parameter group for Repost (Characteristic Relationships) is only a namesake and does not contain any data.

Step by step solution overview

Reference Data

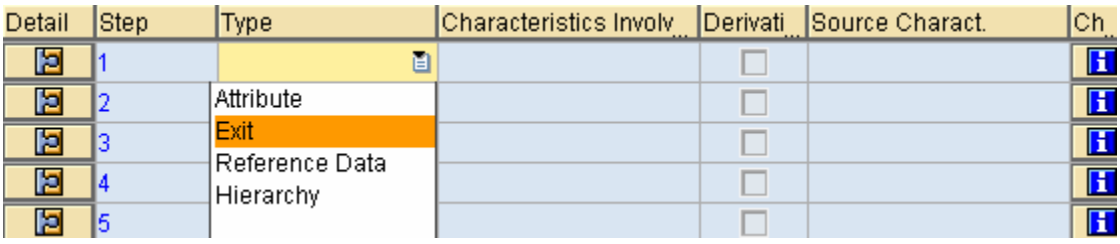
The mapping to derive the profit center from the combination of the Material and CSR Fields is stored in an ODS (ZXSLSMAP) which is loaded from the source system. It also contains the mapping for Sales Office, which we are not using herein. The layout of the ODS is as shown below.



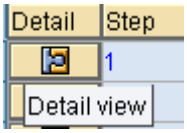
Characteristic Relationship

In order to define a characteristic relationship for the planning area, Click on '***Change planning area'***, and navigate to the tab '***Characteristic Rels'***.

Define the type as '***Exit***'.



Click on '***Detail View'*** to define the details for the Characteristic Relationship.



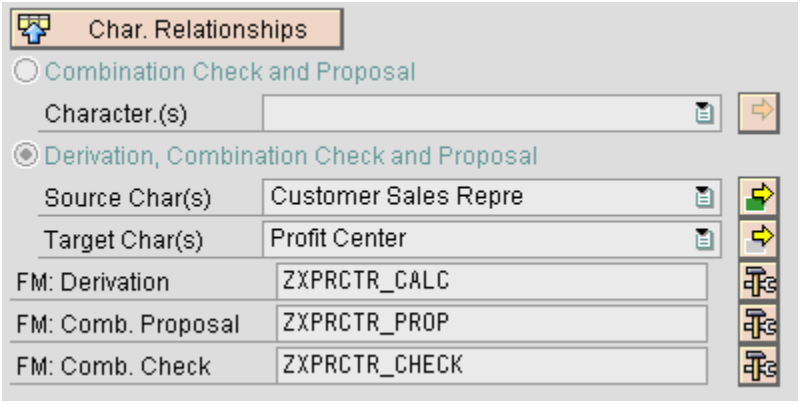
Choose the radio button '***Derivation, Combination Check and Proposal'***.

Enter the source and target characteristics.

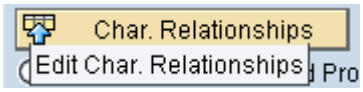
In our case the source characteristics would be '*Material*' and '*Customer Sales Representative'*. The target characteristic would be '*Profit Center'*.

Enter the Function modules to be used for Derivation, Combination Check and Proposal.

In our case, we enter the same as shown below (We shall see the implementation of these function modules later).

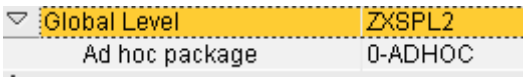


To return back to the definition of further characteristic relationships, click on '***Edit Char. Relationships'***.



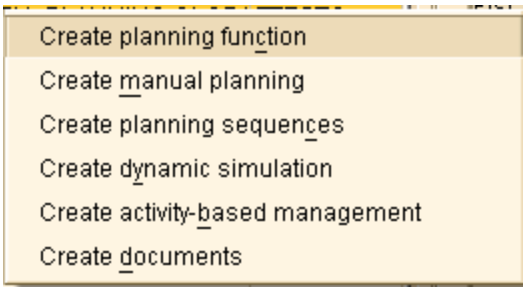
Planning function – Repost (Characteristic Relationship)

Ensure that you have a planning level defined that contains all the characteristics and key figures defined in the planning area.

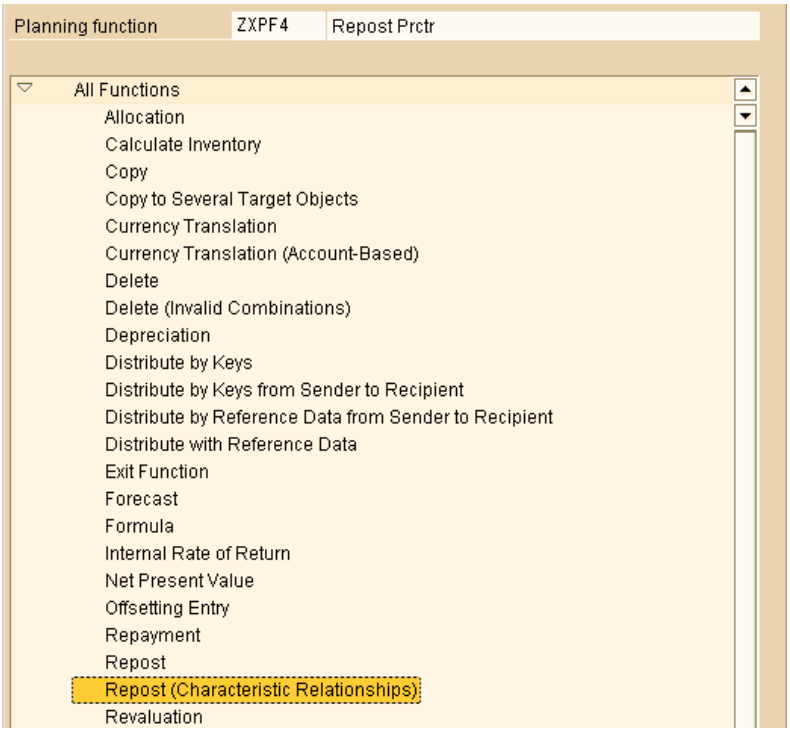


Create a planning package in case you want to place selections on the data.

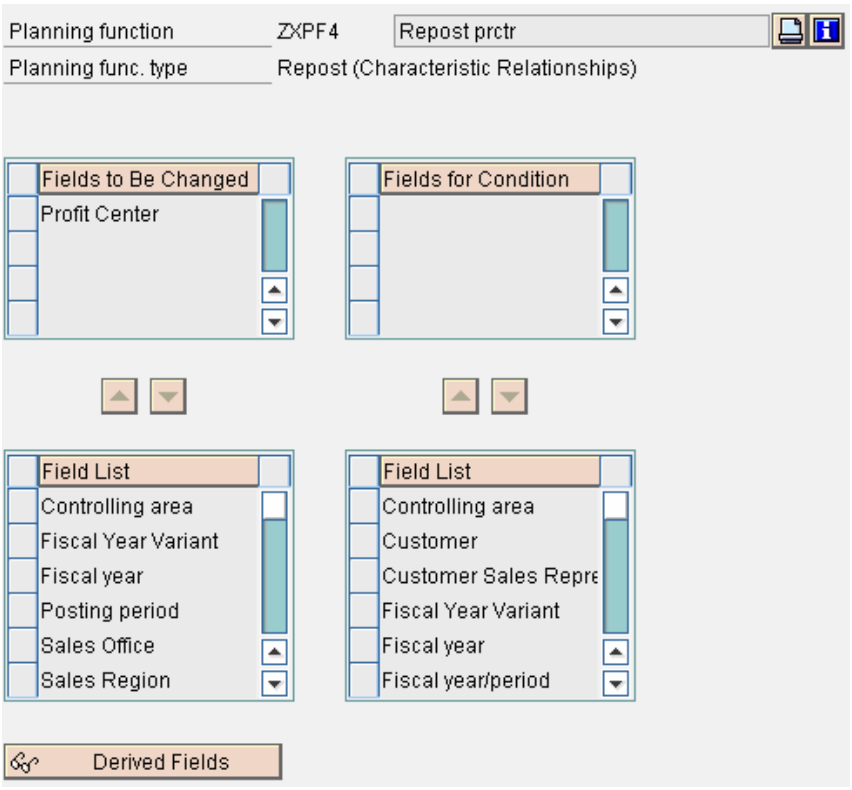
Inside the Planning Level, go to '***Create Planning Function'***.



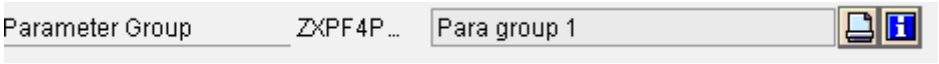
Choose the function type '***Repost (Characteristic Relationships)***'



Select the characteristic to be changed. In our case, we will choose '***Profit Center'***.



Define a Parameter group for the Planning function



Function Modules

Derivation

For derivation, we basically read the values of the fields for Material and CSR and read the corresponding value of Profit Center from the ODS. However, this logic could be much more complicated and proper use of ABAP can help realize the same as well. The code for the function module used is as given below:

```
FUNCTION ZXPRCTR_CALC.
*"----------------------------------------------------------
*"*"Local Interface:
*"  IMPORTING
*"    REFERENCE(I_AREA) TYPE  UPC_Y_AREA
*"    REFERENCE(ITO_CHA) TYPE  UPC_YTO_CHA
*"    REFERENCE(ITO_SOURCE) TYPE  UPC_YTO_CHA
*"    REFERENCE(ITO_TARGET) TYPE  UPC_YTO_CHA
*"  CHANGING
*"    REFERENCE(XS_CHAS) TYPE ANY
*"  EXCEPTIONS
*"    FAILED
*"----------------------------------------------------------

* DERIVATION FOR PROFIT CENTER

*---------------------------------------
* Commented by ADI - included in global
*Tables: /BIC/AZXSLSMAP00
*---------------------------------------

data: profctr like /BIC/ZXSLSMAP00-PROFIT_CTR.

FIELD-SYMBOLS: <material>, <csr>, <prctr>, <lsdata>.

  ASSIGN COMPONENT '0PROFIT_CTR' OF STRUCTURE <lsdata> TO  <prctr>.
  ASSIGN COMPONENT '0MATERIAL' OF STRUCTURE <lsdata> TO <material>.
  ASSIGN COMPONENT 'ZXCSR' OF STRUCTURE <lsdata> TO <csr>.


READ TABLE ITO_CHA TRANSPORTING NO FIELDS
    WITH KEY chanm = '0MATERIAL'.

IF SY-SUBRC = 0.
  SELECT SINGLE * FROM /BIC/AZXSLSMAP00
    WHERE /BIC/ZXCSR EQ <csr> AND
    MATERIAL EQ <material>.

  MOVE /BIC/AZXSLSMAP00-PROFIT_CTR to profctr.
ELSE.
  MESSAGE E002(Z00) WITH <material> <csr>.
        RAISING FAILED.
ENDIF.

<prctr> = profctr.

ENDFUNCTION.
```


Proposal

```
FUNCTION ZXPRCTR_PROP.
*"----------------------------------------------------------
*"*"Local Interface:
*"  IMPORTING
*"    REFERENCE(I_AREA) TYPE  UPC_Y_AREA
*"    REFERENCE(ITO_CHA) TYPE  UPC_YTO_CHA
*"    REFERENCE(ITO_CHASEL) TYPE  UPC_YTO_CHASEL
*"  CHANGING
*"    REFERENCE(ETH_CHAS) TYPE  HASHED TABLE
*"  EXCEPTIONS
*"    FAILED
*"----------------------------------------------------------

* PROPOSITION OF PROFIT CENTER

*---------------------------------------
* Commented by ADI - included in global
*Tables: /BIC/AZXSLSMAP00
*---------------------------------------

data: profctr like /BIC/ZXSLSMAP00-PROFIT_CTR.

FIELD-SYMBOLS: <material>, <csr>, <prctr>, <lsdata>.

loop at ETH_CHAS ASSIGNING <lsdata>.
  ASSIGN COMPONENT '0PROFIT_CTR' OF STRUCTURE <lsdata> TO  <prctr>.
  ASSIGN COMPONENT '0MATERIAL' OF STRUCTURE <lsdata> TO <material>.
  ASSIGN COMPONENT 'ZXCSR' OF STRUCTURE <lsdata> TO <csr>.


  SELECT SINGLE * FROM /BIC/AZXSLSMAP00
    WHERE /BIC/ZXCSR EQ <csr> AND
    MATERIAL EQ <material>.

  MOVE /BIC/AZXSLSMAP00-PROFIT_CTR to profctr.

  <prctr> = profctr.

ENDLOOP.

ENDFUNCTION.
```

Combination Check

The Combination Check ensures that the values that are entered in the transaction data are correct. In this case, we are raising an error message thereby enforcing the conformance. We could change this message to a warning as well.

```
FUNCTION ZXPRCTR_CHECK.
*"----------------------------------------------------------
*"*"Local Interface:
*"  IMPORTING
*"    REFERENCE(I_AREA) TYPE  UPC_Y_AREA
*"    REFERENCE(IS_CHAS) TYPE  ANY
*"    REFERENCE(ITO_CHA) TYPE  UPC_YTO_CHA
*"    REFERENCE(ITO_CHASEL) TYPE  UPC_YTO_CHASEL
*"  EXCEPTIONS
*"    INVALID
*"----------------------------------------------------------
```



```
* COMBINATION CHECK FOR PROFIT CENTER

*--------------------------------------
* Commented by ADI - included in global
*Tables: /BIC/AZXSLSMAP00
*--------------------------------------

data: profctr like /BIC/ZXSLSMAP00-PROFIT_CTR.

FIELD-SYMBOLS: <material>, <csr>, <prctr>, <lsdata>.

  ASSIGN COMPONENT '0PROFIT_CTR' OF STRUCTURE <lsdata> TO  <prctr>.
  ASSIGN COMPONENT '0MATERIAL' OF STRUCTURE <lsdata> TO <material>.
  ASSIGN COMPONENT 'ZXCSR' OF STRUCTURE <lsdata> TO <csr>.


  SELECT SINGLE * FROM /BIC/AZXSLSMAP00
    WHERE /BIC/ZXCSR EQ <csr> AND
    MATERIAL EQ <material>.

  MOVE /BIC/AZXSLSMAP00-PROFIT_CTR to profctr.

IF <prctr> NE profctr.
  MESSAGE E001(Z00) WITH profctr <prctr>.
        RAISING INVALID.
ENDIF.


ENDFUNCTION.
```


Related Content

Further information on these topics can be found using the links below :

Characteristic Relationships

Repost (Characteristic Relationships)

Repost

For a previous publication on using ABAP in Planning functions in BW-BPS, click on the link below :

Accessing BW Masterdata in BPS using Exit Functions

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.

