Using User Exit for Variables in BEx Reporting

Applies to:
SAP BI system & BEx. For more information, visit the Business Intelligence homepage.

Summary
To describe the process followed to use a user exit for complex reporting in BEx. This involves a significant amount of ABAP coding

Author:  R. Prem Kumar
Company:  SAP
Created on: 31 May 2009

Author Bio
The author works for SAP. His expertise includes SAP BI and ABAP
Table of Contents

Introduction .........................................................................................................................................................3
Goal.................................................................................................................................................................3
Graphical view ....................................................................................................................................................3
   Scenario 1 ....................................................................................................................................................................3
   RESULT ..........................................................................................................................................................6
   Scenario 2 ....................................................................................................................................................................6
   RESULT ..........................................................................................................................................................7
Related Contents ................................................................................................................................................8
Copyright.............................................................................................................................................................9
**Introduction**

To describe the process followed to use a user exit for complex reporting in BEx. This involves a significant amount of ABAP coding.

**Goal**

This document describes how to use the user exit EXIT_SAPLRRS0_001 for handling variables created with type customer exit in the query designer for a complex BEx report.

**Graphical view**

**Scenario 1**

Let us consider a scenario where a characteristic variable of type customer exit is to be populated with a value based on the value entered by the user in another input variable during selection.

In the following screen shots the scenario considered is, the user would input a date for the variable ZICCALDAY. Inside the user exit, this value needs to be captured and we would have to derive the previous fiscal year and should be passed to the customer exit variable ZVCPREYRCAL.

The variable ZVCPREYRCAL is created on the info-object 0FISCYEAR(Fiscal Year).
Now let's see how to use the user exit EXIT_SAPLRRS0_001.

As a first step, we need to create a project using T-code CMOD.

In this example, a project by the name /ATB/CA0 is created.

Assign the enhancement RSR00001 & click on Components.
Using User Exit for Variables in BEx Reporting

**SAP Enhancements in Enhancement Project /ATB/CA0**

<table>
<thead>
<tr>
<th>Enhancement</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSR00001</td>
<td>BI: Enhancements for Global Variables in Reporting</td>
</tr>
</tbody>
</table>

Choose the function exit **EXIT_SAPLRRS0_001**

**Display /ATB/CA0**

<table>
<thead>
<tr>
<th>Project</th>
<th>Enhancement</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ATB/CA0</td>
<td>Enhancement Proj</td>
<td></td>
</tr>
<tr>
<td>RSR00001</td>
<td>BI: Enhancement</td>
<td></td>
</tr>
</tbody>
</table>

Activate the Project.

Double click on the function exit **EXIT_SAPLRRS0_001**, this will open the function module

**Function Builder: Display EXIT_SAPLRRS0_001**

```
FUNCTION EXIT_SAPLRRS0_001.
  """“Lokale Schnittstelle:"
  "IMPORTING"
  " VALUE(I_VNAM) LIKE RSZGLOBV-VNAM"
  " VALUE(I_VARTYP) LIKE RSZGLOBV-VARTYP"
  " VALUE(I_IOBJNAM) LIKE RSZGLOBV-IOBJNAM"
  " VALUE(I_CS_CMOD_PRO) TYPE RSZGLOBV-CMOD_PRO"
  " VALUE(I_S_RKB1D) TYPE RSZGLOBV-RKB1D"
  " VALUE(I_PERIV) TYPE RSZGLOBV-RA001-S_RKB1F-PERIV"
  " VALUE(I_T_VAR_RANGE) TYPE RSZGLOBV-RA001-S_RKB1F-PERIV"
  " VALUE(I_STEP) TYPE I DEFAULT 0"
  " EXPORTING"
  " VALUE(E_T_RANGE) TYPE RSZGLOBV-RKB1F-PERIV"
  " VALUE(E_MEEHT) LIKE RSZGLOBV-MEEHT"
  " VALUE(E_NEFAC) LIKE RSZGLOBV-NEFAC"
  " VALUE(E_WAERS) LIKE RSZGLOBV-WAERS"
  " VALUE(E_WHFAI) LIKE RSZGLOBV-WHAIS"
  " CHANGING"
  " VALUE(C_S_CUSTOMER) TYPE RSZGLOBV-RKB1F-PERIV"
  " INCLUDE ZXRSRU01 .
```

Please note that, alternatively once the project is activated with the required function exit, this can also be displayed in the t-code for Function modules **se37**

Double click on the include **ZXRSRU01**, to open the editor where the required coding can be done

In the above function module (exit) the import parameter **I_VNAM** holds the variable (i.e name of the variable)

The import parameter **I_T_VAR_RANGE** holds all the variables and its values that were created in the BEx report
The import parameter `I_STEP` can hold values 1 or 2 or 3. This is a very important parameter, as this identifies, the step in which the required variable is to be processed,

If `I_STEP` = 1, the variable needs to be processed before the report’s selection screen appears, an example for this would be like, if some default values are to be populated to the selection screen variables before user input, `I_STEP` = 1 can be used. This would be explained in scenario 2.

If `I_STEP` = 2, the variable is to be processed after the report’s selection appears (ie after user input). An example for this step is explained in scenario 1 of this document.

`I_STEP` = 3, is used for validation purpose, error messages can be raised here.

The export parameter `E_T_RANGE` is the one to which the calculated value needs to be passed.

Using the ABAP statements `CASE, ENDCASE & WHEN` the code for the desired variable can be executed.

See below the example code:

```abap
DATA: LS_T_VAR_RANGE TYPE RRS0_S_VAR_RANGE,
     LV_CALDAY TYPE SY-DATUM,
     LV_FISCYEAR like T009B-BDATJ
CASE i_vnam.
  WHEN 'ZVCPREYRCAL'.
    IF i_step = 2.
      READ TABLE i_t_var_range INTO LS_T_VAR_RANGE
      WITH KEY vnam = 'ZICCALDAY'.
      LV_CALDAY = LS_T_VAR_RANGE-high.
      CALL FUNCTION 'DATE_TO_PERIOD_CONVERT'
        EXPORTING
          I_DATE = LV_CALDAY
          I_MONMIT = '00'
          I_PERIV = 'V3'
        IMPORTING
          E_BUPER = LV_POSTPER
          E_GJAHR = LV_FISCYEAR.
      LV_FISCYEAR = LV_FISCYEAR - 1.
      ls_range-low = LV_FISCYEAR .
      ls_range-opt = 'EQ'.
      ls_range-sign = 'I'.
      APPEND ls_range TO e_t_range.
    ENDIF.
  WHEN 'Variable1'
  WHEN 'variable2'
ENDCASE.
```

Result

Based on the code written above, when the user inputs a date in the variable ZICCALDAY and executes the report, previous fiscal year is calculated and is passed to the variable ZVCPREYRCAL.

Scenario 2

Let us consider a scenario where a characteristic variable of type customer exit & ready for input is to be populated with a default value when the report is executed & when the selection screen appears.

In the following sample code the scenario considered is, when the selection screen appears for a report the variable ZICCALDAY should have a default value of yesterday’s date. Please note to achieve this `I_STEP` = 1 should be used, as described earlier.
See below the example code:

```plaintext
DATA: LV_CALDAY TYPE SY-DATUM.
CASE i_vnam.
  WHEN 'ZICCALDAY'.
    IF i_step = 1.
      LV_CALDAY = SY-DATUM - 1.
      ls_range-low = '00000000'.
      ls_range-high = LV_CALDAY.
      ls_range-opt = 'EQ'.
      ls_range-sign = 'I'.
      APPEND ls_range TO e_t_range.
    ENDIF.
  ENDWHEN.
ENDCASE.
```

**Result**

Based on the code written above, when the selection screen appears the variable ZICCALDAY would have a default value of SY-DATUM – 1.

Please note, similar to the steps explained above, formula variables of type customer exit & text variables of type customer exit can also be used.
Related Contents
For more information, visit the Business Intelligence homepage.