

Variable Exit in Sap BI 7.0 - How to Start



Applies to:

This article is applicable to SAP BI 7.0. For more information, visit the [Business Intelligence homepage](#).

Summary

This document provides an introduction for the creation and management of variables type customer exit in BW queries.

Author: Ricardo de Azeredo Rosa

Company: SAP Labs Brazil

Created on: 24 May 2009

Author Bio



Ricardo is a Support Consultant in SAP Labs Brazil since 2007. He has worked extensively on SAP R/3 and BW as a PM Certified and BW Consultant since 2002. His knowledge includes PM, CS, DMS, PP, PS modules and BW systems. Since 2004 he has focused his knowledge in Business Intelligence (BW, SEM and Planning) working in various critical projects in this area.

Table of Contents

Variable Exit – Starting	3
Project Management of SAP Enhancements	3
BEX Query Designer and Variable Type Customer Exit.....	4
Customer Exit and ABAP Code	5
Testing and Debugging	6
Additional Information	8
Related Contents	9
Copyright.....	10

Variable Exit – Starting

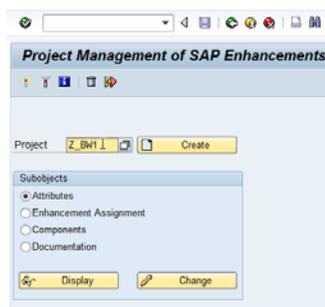
SAP provides a variable of type customer exit for BW queries which customer is able to manage its values depends on their business requirements. It is necessary some background in ABAP to start in this area, the proposal of this document is to help on those tasks. It is structured in the following sequence:

- Project management of SAP Enhancements
- Query BEX Query Designer and Variable Type Customer Exit
- Customer Exit and ABAP Code
- Test and Debug
- Additional Information

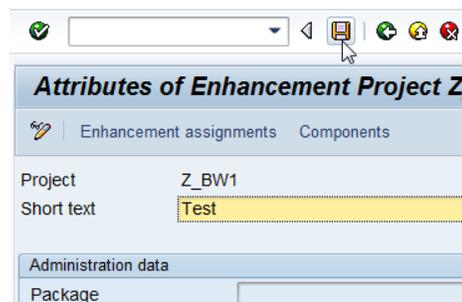
Project Management of SAP Enhancements

In order to start it is necessary to activate a new project for a SAP enhancement. So as to execute that you need to call the transaction CMOD, inform a name for your project and press the button 'create'. A second screen requesting a short description will be showed, just enter the required information and 'save':

1



2



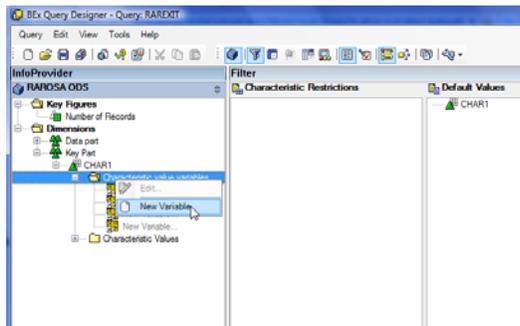
It is necessary to inform the enhancement for your new project. In the list, select that one responsible for handle the variables in the BI/BW queries: 'RSR0001 – BI: Enhancements for Global Variables in Reporting'

Choosing 'components', it is possible to verify that function module EXIT_SAPLRRS0_001 is already assigned for the new project, selecting that one (by double clicking), system jumps to the function module definition which has basically an instruction like 'INCLUDE ZXRSRU01.'. Actually the report ZXRSRU01 is reserved for store the customer code which will manage the variable values in the BW query.

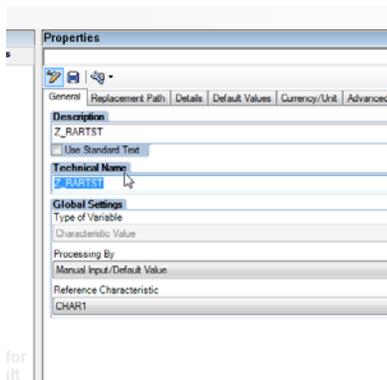
Now it is necessary to create the report ZXRSRU01 in transaction SE38. But before this procedure we need a variable which will be handled in that report. In the next item the steps to create a simple query and a user exit variable type will be explained.

BEX Query Designer and Variable Type Customer Exit

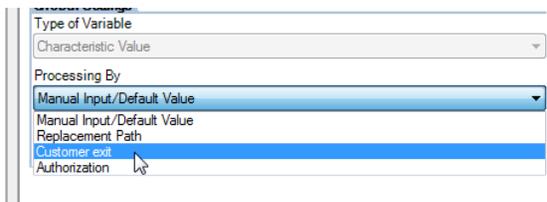
1. In the query designer 7.1, create a new query following the standard procedure. Select the characteristic which you want to filter and create a new variable:



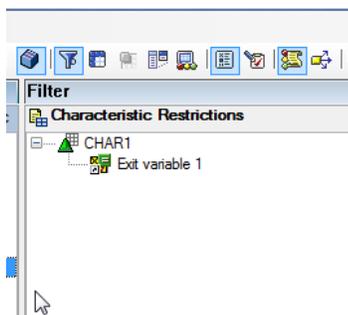
2. Select the technical name and the description for your new variable in the appropriated area:



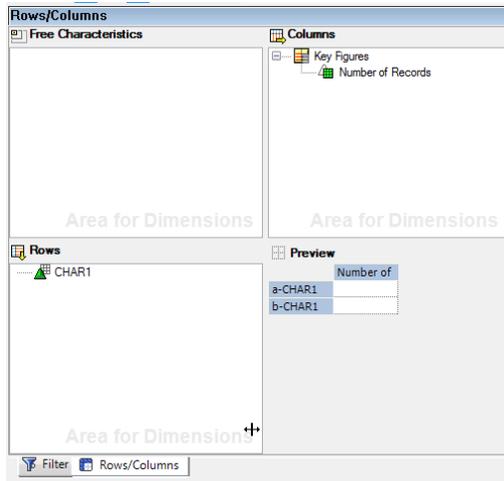
3. Set the type 'Customer exit' in the field 'Processing by':



4. Select your characteristic and restrict that one by the variable created in the previous item as a filter for your query (use the drag and drop):



5. Create a simple query in the 'Rows/Columns' area that we can use as example:



6. Save your query and now we are able to create the customer exit in the program ZXRSRU01.

Customer Exit and ABAP Code

Now the program ZXRSRU01 can be created, you can use the transaction SE38 for that.

The most important information here is that the 'customer exit program' will receive the variable name of the BW query in the ABAP variable 'i_vnam', which is defined as an import parameter in the function module EXIT_SAPLRRS0_001 and the BW query's variable value is handled in the table E_T_RANGE.

So: it is necessary to create a program that check the variable name in order to confirm if we are handling the correct one and manage its current values in the table E_T_RANGE.

It is an example of the code:

```

*&-----*
*& Include          ZXRSRU01
*&-----*
DATA: l_s_range TYPE rrrangesid.
CASE i_vnam.
  WHEN 'Z_RARTST' .
    l_s_range-low = '12345' .
    l_s_range-sign = 'I' .
    l_s_range-opt = 'EQ' .
    APPEND l_s_range TO e_t_range.
ENDCASE .

```

You can create a similar program for a test proposes. Just be careful with the name of the variable, it should be the same you have informed in the restriction of your query in the previous item (in this example 'Z_RARTST' – of course the technical name.).

Active and save your new program before leave.

Before to execute the test, you need to go in the transaction CMOD once more and activate the customer exit otherwise the system will not execute your code.

Change Z_BW					
Project				Z_BW	BW
Enhancement	Impl		Exp	RSR00001 BI: Enhancements for Global Variables in Reporting	
Function exit				EXIT_SAPLRRS0_001	

Change Z_BW					
Project				Z_BW	BW
Enhancement	Impl		Exp	RSR00001 BI: Enhancements for Global Variables in Reporting	
Function exit				EXIT_SAPLRRS0_001	

Your Customer Exit is ready to be tested. Transaction RSRT is a good option for that:

Testing and Debugging

Note: This is not the idea of this document to discuss about ABAP debug; maybe some knowledge in this area is required

First, set a break point in your code, for instance, as per example below:

```

Include ZXRSRU01 Inactive
*&-----
*& Include ZXRSRU01
*&-----
DATA: l_s_range TYPE rrrangesid.
BREAK-POINT.
CASE i_vnam.
  WHEN 'Z_RARTST'.
    l_s_range-low = '12345'.
    l_s_range-sign = 'I'.
    l_s_range-opt = 'EQ'.
    APPEND l_s_range TO e_t_range.
ENDCASE.

```

Call transaction RSRT and execute your query:

The screenshot shows the SAP Query Monitor interface. At the top, there are several buttons: 'Execute', 'Execute + Debug', 'Generate Report', 'Properties', 'Messages', and 'Help Te'. Below these are 'Perform Execute', 'Technical Information', 'Cache Monitor', 'Query Variants', and 'IGS Test'. The 'Query' field is populated with 'RAR0SA1 / RAREXIT'. Below the query field, there is a 'Query Display' dropdown menu set to 'List' and two radio buttons for 'Parameter 1' (selected) and 'Parameter 2'.

Additional Information

Of course this is just a simple example for starting with this feature. Some additional knowledge is required for deeper developments in this area.

In the debug mode you are able to explore some other parameters like I_STEP and I_S_RKB1D for instance. Checking I_S_RKB1D, just as an example, you have access to the query attribute, it means that beyond the variable name it is possible to check some other information in order to set the variable value, like query and infoprovider name and so on. Checking the I_STEP, as another example, it will help you to set the time that exit is triggered, as following:

I_STEP = 1

Call directly before the variable is entered

I_STEP = 2

Call directly after the variable is entered. This step is activated only if the same variable could not be filled for

I_STEP = 1 and it is not ready for input.

I_STEP = 3

In this call, you can check the values of the variables.

Activating an exception (RAISE) causes the variable screen to appear again. I_STEP = 2 is also then executed once more.

I_STEP = 0

The enhancement is not called from the variable screen. The call can come from the authorization check or from the monitor.

Related Contents

http://help.sap.com/saphelp_nw70/helpdata/EN/f1/0a56f5e09411d2acb90000e829fbfe/frameset.htm
[note 492504 - Dependent customer exit-type variables](#)

http://help.sap.com/saphelp_nw70/helpdata/EN/f1/0a56f5e09411d2acb90000e829fbfe/frameset.htm

For more information, visit the [Business Intelligence homepage](#).

Copyright

© Copyright 2009 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects S.A. in the United States and in other countries. Business Objects is an SAP company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.