How to Find the Where Used List of Query Restrictions

Applies to:

SAP BW (3.5) / SAP BI (7.0). For more information, visit the EDW homepage

Summary

This article contains an ABAP report which is used to find out the similar query restrictions used in different queries in system.

Author: Durgesh Gandewar

Company: Infosys Limited

Created on: 11 July 2011

Author Bio

Durgesh Gandewar is working with Infosys Limited and he has got overall experience of 4 years on different implementation projects in SAP ABAP, SAP BI 7.0.

Table of Contents

Business Scenario

There are many scenarios in which we want to find out which are all the queries having specific restrictions.

E.g. Consider a business scenario where I have a multiprovider which contains employee information in three different cubes,

- EMEA
- AMERICAS
- APJ

Over the years data in these cubes has got increased and now I am going to logically partition these cubes in 3 more new cubes, EMEA_HIST, AMERICAS_HIST, and APJ_HIST. In this new scenario all my history cubes will have data till 2010 and the data after FY 2010 till date is available in original cube.

The queries which are based on this multiprovider and has got hard restriction of InfoProvider needs to be changed to accommodate newly created history cube. And we need the list of all such queries which have Infoprovider restriction.

Processing Logic

So as to achieve the above requirement I have built an ABAP report. This report will give us required information with below given processing logic,

- All the information related to query restrictions is stored in table RSZRANGE so based on the filter criteria we will select the appropriate restriction UUID's from this table.
- Now we need to link this UID's with BEX reporting component UID's this linking information can be found in RSZELTXREF table.
- After that finally we need to put all these UID's and do the comparison with COMPID's of main reporting table RSRREPDIR.
- Once we get the required reports we will fetch the report description from RSZELTTXT table.

Tables Used

RSZRANGE:

Table name	RSZRANGE	
Description	Selection specification for an element	
Useful fields	***Field Name***	***Description***
	ELTUID	BW: Unequivocal ID in room and time

RSZELTXREF:

Table name	RSZELTXREF	
Description	Selection specification for an element	
Useful fields	***Field Name***	***Description***
	SELTUID	UUID in compressed form

RSRREPDIR:

Table name	RSRREPDIR	
Description	Directory of all reports	
Useful fields	***Field Name***	***Description***
	COMPUID	UUID in compressed form
	INFOCUBE	Name of InfoProvider
	COMPID	Query Name
	AUTHOR	Owner of Query
	LAST USER	Last changed by

RSZELTTXT:

Table name	RSZELTTXT	
Description	Texts of reporting component elements	
Useful fields	***Field Name***	***Description***
	ELTUID	Unequivocal ID in room and time
	TXTLG	Query Description

Below given flow will help us to understand the linking between these tables.

Data Browser: Table RSZRANGE Select Entries 30

Table: RSZRANGE
Displayed Fields: 7 of 7 Fixed Columns: 1 List Width 0200

ELTUID	OBJVERS	IOBJNM	SIGN	OPT	LOW
0F264TM5IT1Y9RW6L1MK3N71Q	A	0INFOPROV	I	EQ	PGFSWBS2
0RUJ0TDPR3HZ4SZJUZVP5L20E	A	0INFOPROV	I	EQ	PGFSWBS2
1HF8SSWU7OE0UV6AEWDZ9GRXQ	A	0INFOPROV	I	EQ	PGFSWBS2
1U7LOSOEFYU1PW9NOUN4BEMWE	A	0INFOPROV	I	EQ	PGFSWBS2
2DE50SBQSEI30FWPLS0TWBFCE	A	0INFOPROV	I	EQ	PGFSWBS2
2O1RRG4PNB7FQATIO2KG3XN5A	A	0INFOPROV	I	EQ	PGFSWBS2
378B3FS1ZQVH0UGKKZY5OUFLA	A	0INFOPROV	I	EQ	PGFSWBS2

Data Browser: Table RSZELTXREF Select Entries

Table: RSZELTXREF
Displayed Fields: 3 of 3 Fixed Columns: 3 List

SELTUID	OBJVERS	TELTUID
3SV6ILLS4EHOWQOM5WXIOJCXT	A	0RUJ0TDPR3HZ4SZJUZVP5L20E
3SV6ILTGND3EFD82BQZUYLBNL	A	0RUJ0TDPR3HZ4SZJUZVP5L20E
3SV6IM156BP3XZRIHL278NADD	A	0RUJ0TDPR3HZ4SZJUZVP5L20E
7EJVM107DUH381EFORABHT3ZI	A	1HF8SSWU7OE0UV6AEWDZ9GRXQ
7GOLRCYSR87RD7KZWEZUC4R5A	A	1U7LOSOEFYU1PW9NOUN4BEMWE
7ITBWOXE4LYFIDRK42PD6GEB2	A	2DE50SBQSEI30FWPLS0TWBFCE

Data Browser: Table RSRREPDIR Select Entries 5

Check Table...

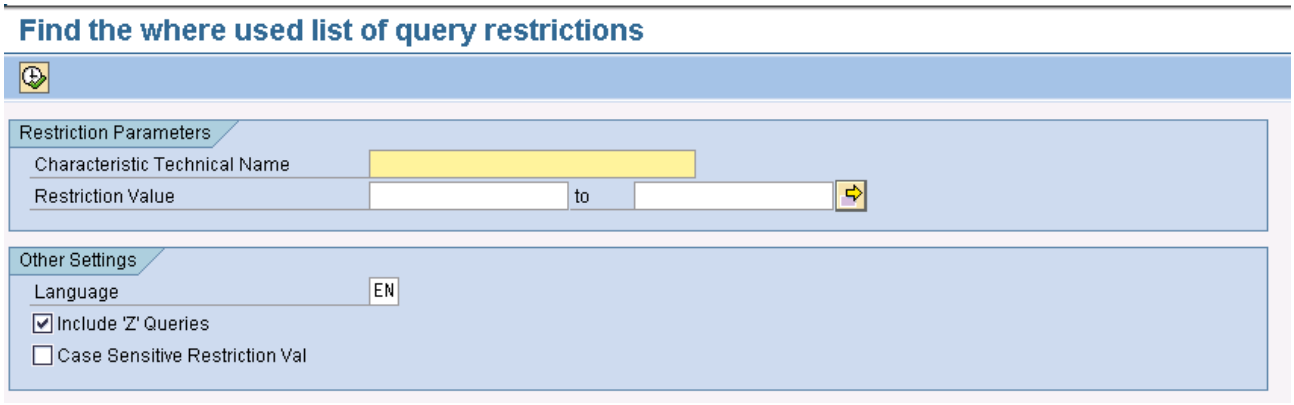
Table: RSRREPDIR
Displayed Fields: 5 of 5 Fixed Columns: 3 List Width 0250

COMPUID	INFOCUBE	COMPID	AUTHOR	LASTUSER
0IDPAZ3N3LHM8KYOKCRMM7BZI	PGFSGCWC2	ZTWBSCOSTS1	RSAHA	RSAHA
3T8HABU078R2EMRJNXI0X63I9	PGFSGCPR5	PGFSUSPR5_DASH_PRT_Q1	C00300000168	90018308
3VQA2410M2E1GCBOEU82WL5QA	PGFSGCPR5	PGFSUSPR5_DASH_PRT_Q1_TEST	C00300000168	C00300000165
3VQAJ04PPH060S16CZK6W3IGY	PGFSGCPR5	PGFSUSPR5_DASH_CTRL_TEST	C00300000165	C00300000165
3VTSWC6HIRMYAYPY0SQEKVBK2	PGFSGCPR5	ZWW_PGFSUSPR5_DASH_PRT_Q1_1	C00300000298	C00300000298

And Finally COMPUID of RSRREPDIR needs to be matched with RSZELTTXT to get query description.

How to use the Program

Once you execute the below given program you will see following selection screen,



Parameters Used

Characteristics Technical Name: This is technical name of Infoobject on which query restriction is based eg. 0Infoprov, 0comp_code etc.

Restriction Value: This is the actual value of restriction e.g. 0InfoProv = APJ, 0Comp_code = APJ01 etc.

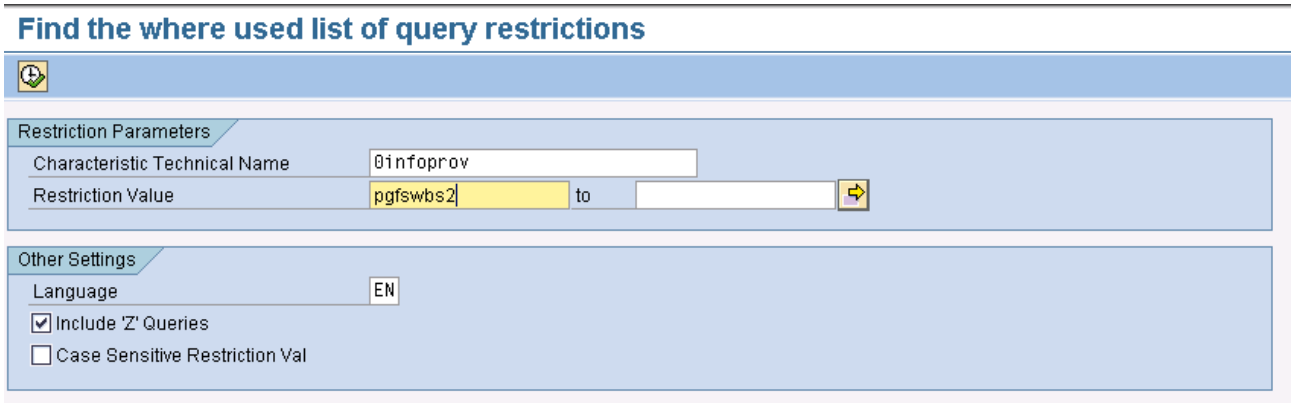
Language: The Query description will be shown in the language input given by user.

Include 'Z' Queries: Whether you want to include 'Z' queries in output or not.

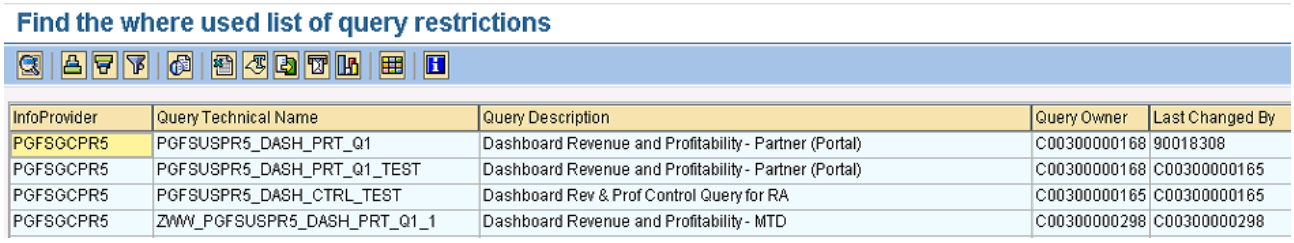
Case Sensitive Restriction Val: By default all the restriction values entered are converted to upper case. If we want to maintain the case sensitivity then we will have to mark this flag.

Example

In our scenario I want to find out all the queries which have hard restriction of infoprovider PGFSWBS2.



Output is displayed as follows,



InfoProvider	Query Technical Name	Query Description	Query Owner	Last Changed By
PGFSGCPR5	PGFSUSPR5_DASH_PRT_Q1	Dashboard Revenue and Profitability - Partner (Portal)	C00300000168	90018308
PGFSGCPR5	PGFSUSPR5_DASH_PRT_Q1_TEST	Dashboard Revenue and Profitability - Partner (Portal)	C00300000168	C00300000165
PGFSGCPR5	PGFSUSPR5_DASH_CTRL_TEST	Dashboard Rev & Prof Control Query for RA	C00300000165	C00300000165
PGFSGCPR5	ZWW_PGFSUSPR5_DASH_PRT_Q1_1	Dashboard Revenue and Profitability - MTD	C00300000298	C00300000298

Source Code

See the below attached source code.

```
*&---------------------------------------------------------------------*
*& Report  ZBW_QUERY_RESTRICTION
*&
*&---------------------------------------------------------------------*
*&
*& TITLE: Find the where used list of query restrictions
*&
*& CREATED ON: 11 July 2011
*&
*& CREATED BY: Durgesh Gandewar
*&
*& DESCRIPTION: This program can be used to find the characteristics
*& value restriction getting used in BEX Queries. It will list all BEX
*& queries with common characteristic restriction value.
*&
*&---------------------------------------------------------------------*

REPORT  zbw_query_restriction.

TABLES: rszrange, rszelttxt.

************************ Type Declarations ****************************

TYPE-POOLS: slis.

TYPES:

BEGIN OF ty_rsrrepdir,
compuid       TYPE         sysuuid_25, "Compressed format UUID
infocube      TYPE         rsinfocube, "Infoprovider name
compid        TYPE         rszcompid,  "Query Name
author        TYPE         xuauthor,   "Query Owner
lastuser      type         LAST_USER,  "Last changed by
END OF ty_rsrrepdir,

BEGIN OF ty_rszrange,
eltuid        TYPE         rsuniidx,   "Unequivocal ID in room and time
END OF ty_rszrange,

BEGIN OF ty_rszeltxref,
seltuid       TYPE         sysuuid_25, "UUID in compressed form
END OF ty_rszeltxref,

BEGIN OF ty_rszelttxt,
eltuid        TYPE         sysuuid_25,
txtlg         TYPE         rstxtlg,
END OF ty_rszelttxt,

BEGIN OF ty_range,
sign          TYPE         raldb_sign,
option        TYPE         rsz_operator,
low           TYPE         rsiobjnm,
high          TYPE         rsiobjnm,
```

```
END OF ty_range,

BEGIN OF ty_output,
infocube      TYPE         rsinfocube, "Infoprovider name
compid        TYPE         rszcompid,  "Query Technical Name
txtlg         TYPE         rstxtlg,    "Query Description
author        TYPE         xuauthor,   "Query Owner
lastuser      type         LAST_USER,  "Last Changed by
END OF ty_output.


************************ Internal Tables *******************************

DATA: it_fieldcat    TYPE     slis_t_fieldcat_alv,
      it_output      TYPE STANDARD TABLE OF    ty_output,
      it_range       TYPE STANDARD TABLE OF    ty_range,
      it_rszrangetmp TYPE STANDARD TABLE OF    ty_rszrange,
      it_rszeltxref  TYPE STANDARD TABLE OF    ty_rszeltxref,
      it_rsrrepdir   TYPE STANDARD TABLE OF    ty_rsrrepdir,
      it_rszelttxt   TYPE STANDARD TABLE OF    ty_rszelttxt,
      it_rszrange    TYPE STANDARD TABLE OF    ty_rszrange.


************************ Work Areas ************************************

DATA: wa_fieldcat    TYPE      slis_fieldcat_alv,
      wa_output      TYPE      ty_output,
      wa_range       TYPE      ty_range,
      wa_rszeltxref  TYPE      ty_rszeltxref,
      wa_rsrrepdir   TYPE      ty_rsrrepdir,
      wa_rszelttxt   TYPE      ty_rszelttxt,
      wa_rszrange    TYPE      ty_rszrange.

************************ Variables *************************************
DATA:

i_title_grid    TYPE   lvc_title,
i_repid  TYPE   sy-repid.


*----------------------------------------------------------------------*
*               SELECTION SCREEN DETAILS
*----------------------------------------------------------------------*

SELECTION-SCREEN BEGIN OF BLOCK blk1 WITH FRAME TITLE text-001.

PARAMETER p_char TYPE rsiobjnm OBLIGATORY.
SELECT-OPTIONS  s_val FOR rszrange-low.

SELECTION-SCREEN END OF BLOCK blk1.



SELECTION-SCREEN BEGIN OF BLOCK blk2 WITH FRAME TITLE text-002.

PARAMETER p_lang TYPE sylangu DEFAULT 'E'.
```

```
PARAMETER p_zinclu TYPE char1 AS CHECKBOX DEFAULT 'X'.
PARAMETER p_case   TYPE char1 AS CHECKBOX.

SELECTION-SCREEN END OF BLOCK blk2.


************************************************************************
*                SELECTION OF DATA                                     *
************************************************************************

* get the initial UID's based on selection screen values.

it_range = s_val[].

IF it_range IS INITIAL.

  SELECT eltuid
         FROM rszrange
         INTO TABLE it_rszrange
         WHERE objvers = 'A' AND
               iobjnm = p_char.
ELSE.

  LOOP AT it_range INTO wa_range.

* Check for case sensitivity.
    IF p_case IS INITIAL.
      TRANSLATE wa_range-low TO UPPER CASE.
      TRANSLATE wa_range-high TO UPPER CASE.
    ENDIF.

    SELECT eltuid
           FROM rszrange
           INTO TABLE it_rszrangetmp
           WHERE objvers = 'A' AND
                 iobjnm = p_char AND
                 sign = wa_range-sign AND
                 opt  = wa_range-option AND
                 low  = wa_range-low AND
                 high = wa_range-high.

    IF sy-subrc = 0.
      APPEND LINES OF it_rszrangetmp TO it_rszrange.
    ENDIF.

  ENDLOOP.
ENDIF.


IF it_rszrange IS INITIAL.
  EXIT.
ENDIF.

*Remove duplicates.
SORT it_rszrange BY eltuid.
DELETE ADJACENT DUPLICATES FROM it_rszrange COMPARING eltuid.
```

```
*Now get the reference ID's for this unique eltuid's.
SELECT seltuid
       FROM rszeltxref
       INTO TABLE it_rszeltxref
       FOR ALL ENTRIES IN it_rszrange
       WHERE objvers = 'A' AND
       teltuid = it_rszrange-eltuid.
IF sy-subrc = 0.
* Now get the actual reports from reporting table.

  SELECT  compuid
          infocube
          compid
          author
          lastuser
  FROM rsrrepdir
  INTO TABLE it_rsrrepdir
  FOR ALL ENTRIES IN it_rszeltxref
  WHERE objvers = 'A' AND
  comptype = 'REP' AND
  compuid = it_rszeltxref-seltuid.
  IF sy-subrc = 0.
* get the query description for all found queries.

    SELECT eltuid
           txtlg
    FROM   rszelttxt
    INTO TABLE it_rszelttxt
    FOR ALL ENTRIES IN it_rsrrepdir
    WHERE objvers = 'A' AND
    langu = p_lang AND
    eltuid = it_rsrrepdir-compuid.
  ENDIF.
ENDIF.

* prepare the output table for ALV display.
LOOP AT it_rsrrepdir INTO wa_rsrrepdir.

  CLEAR: wa_output,wa_rszeltxref.
  MOVE-CORRESPONDING wa_rsrrepdir TO wa_output.

* get the query description.
  READ TABLE it_rszelttxt INTO wa_rszelttxt WITH KEY
     eltuid = wa_rsrrepdir-compuid.

  IF sy-subrc = 0.
    wa_output-txtlg = wa_rszelttxt-txtlg.
  ENDIF.

  APPEND wa_output TO it_output.
ENDLOOP.

* Exclude 'Z' queries from output.
IF p_zinclu IS INITIAL.
  DELETE it_output WHERE compid CP 'Z*'.
```

```
ENDIF.


* Prepare field catalog for ALV display
PERFORM fieldcat.

* Display ALV
PERFORM display.

*&---------------------------------------------------------------------*
*&       Form   FEILDCAT
*&---------------------------------------------------------------------*
*  FIELD  CATALOG TO DISPLAY COLUMNS
*----------------------------------------------------------------------*

FORM  fieldcat .

  wa_fieldcat-col_pos  = 1.
  wa_fieldcat-fieldname = 'INFOCUBE'.
  wa_fieldcat-reptext_ddic = 'InfoProvider'.
  wa_fieldcat-outputlen = 15.
  wa_fieldcat-tabname  = 'IT_OUTPUT'.
  APPEND wa_fieldcat TO it_fieldcat.

  CLEAR  wa_fieldcat.
  wa_fieldcat-col_pos  = 2.
  wa_fieldcat-fieldname = 'COMPID'.
  wa_fieldcat-reptext_ddic = 'Query Technical Name'.
  wa_fieldcat-outputlen = 35.
  wa_fieldcat-tabname  = 'IT_OUTPUT'.
  APPEND wa_fieldcat TO it_fieldcat.

  CLEAR  wa_fieldcat.
  wa_fieldcat-col_pos = 3.
  wa_fieldcat-fieldname = 'TXTLG'.
  wa_fieldcat-reptext_ddic = 'Query Description'.
  wa_fieldcat-outputlen = 60.
  wa_fieldcat-tabname  = 'IT_OUTPUT'.
  APPEND wa_fieldcat TO it_fieldcat.

  CLEAR  wa_fieldcat.
  wa_fieldcat-col_pos = 4.
  wa_fieldcat-fieldname = 'AUTHOR'.
  wa_fieldcat-reptext_ddic = 'Query Owner'.
  wa_fieldcat-outputlen = 12.
  wa_fieldcat-tabname  = 'IT_OUTPUT'.
  APPEND wa_fieldcat TO it_fieldcat.

  CLEAR  wa_fieldcat.
  wa_fieldcat-col_pos = 5.
  wa_fieldcat-fieldname = 'LASTUSER'.
  wa_fieldcat-reptext_ddic = 'Last Changed By'.
  wa_fieldcat-outputlen = 15.
  wa_fieldcat-tabname  = 'IT_OUTPUT'.
  APPEND wa_fieldcat TO it_fieldcat.
```

```
ENDFORM.


*&---------------------------------------------------------------------*
*&       Form   DISPLAY
*&---------------------------------------------------------------------*
*  DISPLAY OUTPUT USING ALV GRID
*----------------------------------------------------------------------*

FORM  display .

  CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
    EXPORTING
      i_callback_program = i_repid
      i_grid_title       = i_title_grid
      it_fieldcat        = it_fieldcat
    TABLES
      t_outtab           = it_output
    EXCEPTIONS
      program_error      = 1
      OTHERS             = 2.
  IF sy-subrc <> 0.

  ENDIF.

ENDFORM.
```

Related Content

Important Tables in SAP BI (NW2004)

http://www.help.sap.com

For more information, visit the EDW homepage

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.