Sreehari V Pillai

March 21, 2018   3 minute read

Handling SELECT OPTIONS in CDS using Table Function

| Follow | | RSS feed | | Like |

10 Likes   22,043 Views   19 Comments

Introduction

This quick blog describes how to push down select options feature of ABAP layer to CDS using Table functions. Before reading this short blog, I would suggest you to go through AMDP-Handling select options

We have been seeing many questions in SCN and other forums regarding how to use select options in AMDP , CDS etc. SELECT OPTIONS is not an SQL feature . Its an ABAP (or OPEN SQL) feature which SQL script does not understand . So, thinking to push this select option variable (range) directly to DB layer is not possible .

Implementation

What can we do ? Only filtering technique SQL can understand is using a where clause . In select options ,we have many choices of passing inputs , like EQ , NE , CP , BETWEEN etc. If we can convert these options into a where clause understood by SQL , we can execute it dynamically in the DB layer.

Can I use Select options in CDS ?

Yes you can , but only in the outer layer right after projecting the data , like we do in ABAP. That is, while consuming this CDS from the ABAP layer you can use the range variable . But this is not what we are looking at . We need to push this down to the DB layer .

Let us create a table function first, to fetch the list of materials along with its text .

Name: *	ZDEMO_SEL_OPT
Description: *	Select Options Demo
Original language:	EN

```
@EndUserText.label: 'Select Options Demo'
define table function ZDEMO_SEL_OPT
with parameters sel_opt : abap.char( 1000 )
returns {
  mandt : abap.clnt;
  matnr : matnr;
  maktx : maktx ;
}
implemented by method zcl_adt=>get_material;
```

This table function accepts a string as parameter and returns the material number and text .

In the corresponding implementation, we use **APPLY_FILTER function** available in SQL to apply dynamic where clause. So, in the **sel_opt** parameter of table function, we are supposed to pass a dynamic where clause corresponding to the select options .

```
⊖ CLASS zcl_adt DEFINITION
      PUBLIC
      FINAL
      CREATE PUBLIC .

      PUBLIC SECTION.
       INTERFACES : if_amdp_marker_hdb.
       CLASS-METHODS : get_material for TABLE FUNCTION ZDEMO_SEL_OPT.
      PROTECTED SECTION.
      PRIVATE SECTION.
  ⌊ ENDCLASS.

  ⊖ CLASS zcl_adt IMPLEMENTATION.
  ⊖ method get_material BY DATABASE FUNCTION FOR HDB LANGUAGE  SQLSCRIPT OPTIONS READ-ONLY using mara makt.

   lt_mara  = apply_filter( mara , :sel_opt );

   return select mara.mandt , mara.matnr , makt.maktx
        from :lt_mara mara inner join makt makt
       on mara.mandt = makt.mandt and
          mara.matnr = makt.matnr and
          makt.spras = 'E';
   ENDMETHOD.
   ENDCLASS.|
```

Here, the filter is applied on mara table and the returned result set is saved in a variable to use further.

Now the ABAP side.

```
data : l_matnr type matnr,
        lt_tab type TABLE OF zdemo_sel_opt,
        L_WHERE TYPE STRING.
SELECT-OPTIONS : s_matnr for l_matnr.



START-OF-SELECTION.
```

```abap
    L_WHERE = CL_SHDB_SELTAB=>combine_seltabs(
                        EXPORTING it_named_seltabs =
                              value #( ( name = 'MATNR' dref = REF #( s_matnr[] ) ) )
                              iv_client_field = 'MANDT'
                ).


    WRITE L_WHERE.
```

What happening here is ,

The abap program has a selection screen with material number as a select option.

The **CL_SHDB_SELTAB** and it's static method **combine_sel_tabs** returns a dynamic where clause based on the select option variable we pass .

We can specify multiple fields and it''s range variables together . This must be done when all the specified columns are available in the target projection. (We can specify MATNR , ERSDA and MATKL together as all these fields are available in MARA table).

Client column name is an optional parameter. If we pass this field , client field is also added in the dynamic where clause.

I passed 2 material numbers in the selection screen and the generated where clause is


```
test

MANDT = '🔴' AND ( MATNR = '000000000011000643'  OR MATNR = '000000000011001212')
```

Now, we shall pass this value (where clause) to the table function we created.

```
select * from zdemo_sel_opt( sel_opt = @l_where ) into table @lt_tab.
```

and the where clause is executed in the DB layer and data is filtered.

Row	MATNR [C(18)]	MAKTX [C(40)]
1	000000000011000643	
2	000000000011001212	

Now , I made a small change in the SQL Script code in table function. Now the columns projected have new names.

```
lt_mat = select mara.mandt as Client,
                mara.matnr as Material,
                makt.spras as Lang,
                makt.maktx as Description
    from  mara mara inner join makt makt
            on   mara.mandt = makt.mandt and
                 mara.matnr = makt.matnr;

lt_out = apply_filter( :lt_mat , :sel_opt);

return select Client as MANDT , Material as MATNR , Description as MAKTX from :lt_out;
ENDMETHOD
```

Now we are applying the filter on a variable which has fields Client , Material , lang and Description. So , changed the dynamic where clause as

```
L_WHERE = CL_SHDB_SELTAB=>combine_seltabs(
                        EXPORTING it_named_seltabs =
                            value #( ( name = 'Material' dref = REF #( s_matnr[] ) )
                                     ( name = 'lang' dref = REF #( s_spras[] ) )
```

```
                                    )
                    iv_client_field = 'Client'
            ).
```

the **apply_filter** function accepts a table , a view or a table variable in the scope and apply dynamic where clause.

Read my other blog on – mapping abap features with abap on hana

Sreehari

Assigned tags

ABAP Development | SAP HANA | ABAP AMDP | ABAP CDS view | cds table function |

View more...

Related Blog Posts

Hierarchy Evaluation using CDS Table Function
By **Pavan Kumar Reddy** , Feb 24, 2017

How to use AMDP Function implementation for a CDS Table Function as a data source in CDS views
By **Harikumar K** , Feb 24, 2016

ABAP News for Release 7.50 - CDS Table Functions Implemented by AMDP

By **Horst Keller** , Oct 21, 2015

Related Questions

19 Comments

You must be **Logged on** to comment or reply to a post.

Michelle Crapo

March 22, 2018 at 7:51 am

Great read.

Michelle

Like (1)

Sreehari V Pillai | Post author

March 22, 2018 at 8:41 am

Thanks Michelle.

Read my confusing blog here and suggest edits please .

Sree

Like (0)

Timo John

March 26, 2018 at 12:17 pm

Thanks for sharing,

this demonstrates by the way the increased complexity in the ABAP-Areas: Until now we neede Open-SQL which is complex enough.

Now we need to handle: ABAP + CDS; ABAP + AMDP, CDS + AMDP ...

This brings great new fun but will take years to arrive in the implementing majority of ABAPers out there.

Thanks for helping

Like (2)

Sreehari V Pillai | Post author

March 26, 2018 at 1:49 pm

Adding Procedure proxy and external views to the list 🙂

Like (0)

surendra kasetty

May 17, 2018 at 8:47 am

Hello Srihari,

when i execute the table function, i can give only one value, i cannot give multiple values, is it possible to get multiples values on table function?

Thanks,

Surendra Kasetty

Like (0)

Sreehari V Pillai | Post author

May 21, 2018 at 5:12 am

Hi Surendra,

Its said under the section in this blog.

Can I use Select options in CDS ?

Sree

Like (0)

Sudeep Dey

August 21, 2018 at 11:45 am

Hi,

I'm not able to see the table function option.
How to create the table function?

I'm in HANA 1709

Thank you.

Like (0)

Sreehari V Pillai | Post author

September 3, 2018 at 12:02 pm

From your below comment, I am sure now you are able to create. Let me know otherwise.

Sreehari

Like (0)

Sudeep Dey

August 22, 2018 at 6:17 am

Hi,

I'm getting Error in the following lines

lt_mara = apply_filter(mara , :sel_op);

return select mara.mandt , mara.matnr , makt.maktx

The error is source code of this class is incomplete in the lt_mara.

How to resolve the error?

Thank You.

Like (0)

Sreehari V Pillai | Post author

September 3, 2018 at 12:03 pm

can you post the complete code ? Not able to understand the error from the snippet.

Sreehari

Like (0)

Tejaswi Paritala

January 23, 2019 at 9:32 am

Hi Sreehari,

Thanks for sharing.

I am getting below error in AMDP Method.

Error : SQLSCRIPT message: invalid identifier: Z131_CL_AMDP_TEST=>MARA#covw &A0&A1&A2&A3&A4&A5&A6&A7&A8&A9

```
30⊖ method test_sel_opt by DATABASE FUNCTION FOR HDB LANGUAGE SQLSCRIPT options READ-ONLY
31  using mara makt.
32
33  lt_mara = APPLY_FILTER( mara, :sel_opt );
34  RETURN
35   select mara.mandt,
36          mara.matnr,
37          makt.maktx
38          from :lt_mara  mara inner JOIN makt makt
⊗39          on mara.mandt = makt.mandt and
40              mara.matnr = makt.matnr and
41              makt.spras = 'E';
42
43
44  ENDMETHOD.
```

Kindly help.

Like (0)

Sreehari V Pillai | Post author

January 23, 2019 at 9:58 am

change :lt_mara mara to :lt_mara m and refer m . this must be due to ambiguity .

Sreehari

Like (1)

Tejaswi Paritala

January 23, 2019 at 11:14 am

Thanks for helping

Like (1)

Invenio Solutions

March 6, 2019 at 2:39 am

Hello Sreehari,,

Firstly thanks to you , to provide a very nice document which helps to create a table function with select options.

I go through the document but I am giving a error while using select option in report where clause it is giving me an error select option was not bound.

Could you please help me on this.

```abap
DATA :  gv_where(1000) TYPE c.

DATA: gt_qmnotif TYPE STANDARD TABLE OF zqmamdp_qualitynotification,
      gs_qmnotif TYPE zqmamdp_qualitynotification.

SELECTION-SCREEN: BEGIN OF BLOCK b1 WITH FRAME TITLE TEXT-001.
 SELECT-OPTIONS: s_ntyp  FOR gs_qmnotif-qmart   NO INTERVALS NO-EXTENSION,
                 s_plant FOR gs_qmnotif-rm_werks NO INTERVALS NO-EXTENSION,
                 s_date  FOR gs_qmnotif-erdat.

SELECTION-SCREEN: END OF BLOCK b1.

 START-OF-SELECTION.

   PERFORM get_selectoption_param.
   PERFORM get_data.
*&---------------------------------------------------------------------*
*& Form GET SELECTOPTION PARAM
*&---------------------------------------------------------------------*
FORM get_selectoption_param .

  gv_where = cl_shdb_seltab=>combine_seltabs(
                              EXPORTING it_named_seltabs =
                                    VALUE #( ( name = 'ERDAT' dref = REF #( s_date[] ) ) )
                                    iv_client_field = 'MANDT'
                          ).

ENDFORM.
*&---------------------------------------------------------------------*
*& Form GET_DATA
*&---------------------------------------------------------------------*
*& text
*&---------------------------------------------------------------------*
*& -->  p1        text
*& <--  p2        text
*&---------------------------------------------------------------------*
FORM get_data .

  CLEAR: gt_qmnotif[].
  SELECT * FROM zqmamdp_qualitynotification
         WHERE s_selopt     = @gv_where
         INTO TABLE @gt_qmnotif.

ENDFORM.
```

```abap
class ZQMCL_QUALITY_NOTIFICATION_TF definition
   public
   final
   create public .

public section.

   INTERFACES: if_amdp_marker_hdb.
   class-methods GET_QUALITYNOTIFICATIONDETAILS FOR TABLE FUNCTION ZQMAMDP_QualityNotification.
protected section.
private section.
ENDCLASS.



CLASS ZQMCL_QUALITY_NOTIFICATION_TF IMPLEMENTATION.


   method GET_QUALITYNOTIFICATIONDETAILS BY DATABASE FUNCTION FOR HDB LANGUAGE SQLSCRIPT OPTIONS REAI

     lt_date  = APPLY_FILTER ( qmel , :s_selopt );

     return select qmel.mandt,
                   qmel.qmart,
                   qmsm.qmnum,
                   qmel.erdat,
                   qmel.rm_werks
                   from :lt_date qmel inner join qmsm qmsm
                   on qmel.mandt  = qmsm.mandt and
                      qmel.qmnum  = qmsm.qmnum;

   endmethod.
ENDCLASS.
```

```
@EndUserText.label: 'Quality Notification Report'
define table function ZQMAMDP_QualityNotification
with parameters s_selopt      : abap.char( 1000 )
returns {

mandt          : abap.clnt;
qmart          : qmart;
qmnum          : qmnum;
erdat          : erdat;
rm_werks       : rm_werks;

}
implemented by| method zqmcl_quality_notification_TF=>GET_QUALITYNOTIFICATIONDETAILS;
```

Regards,

Puneet

Like (0)

Avanish Joshi

March 8, 2019 at 9:40 pm

Puneet,

I haven't tried this on my system yet, but pass the select options values as

select * from Zqmamdp_qualitynotification(s_selopt = gv_where)

into table @gt_data

Shiladitya Ghosh

November 12, 2019 at 2:15 pm

I'm facing the below error while trying to active the code.

```
16
17 METHOD get_material by database function for hdb language sqlscript options read-only using mara makt .
18                                                          SQLSCRIPT message: identifier must be declared: SEL_OUT  &A0&A1&A2&A3&A4&A5&A6&A7&A8&A9
19 lt_mat = select mara.mandt as Client,
20                 mara.matnr as Material,
21                 makt.spras as Lang,
22                 makt.maktx as Description
23         from mara mara inner join makt makt
24         on mara.mandt = makt.mandt and
25            mara.matnr = makt.matnr;
26    lt_out = apply_filter( :lt_mat, :sel_out);
27    Return Select Client as MANDT , Material as MATNR, Description as MAKTX from :lt_out;
28 endmethod.
29 ENDCLASS.
```

How can I avoid this?

Sreehari V Pillai | Post author

December 25, 2019 at 12:05 pm

Screenshot is blurry . I am bit late to comment, I hope you must have fixed the issue by now.

Sree

Sivakumar Arunachalam

December 16, 2019 at 6:19 pm

Hi Sreehari,

I'm facing time out error for this simple CDS with Table Function. Can you please help

CDS with Table Function:

—————————————–

```
@EndUserText.label: 'CDS view with Table function for PRCD_Elements'
define table function YCDS_PRCD_ELEMENTS
with parameters sel_opt : abap.char( 1000 )
returns {
client :mandt;
knumv :knumv;
kposn :kposn;
kschl :kscha;
kbetr :vfprc_element_amount;
kpein :kpein;
kmein :kmein;
knumh :knumh;
kwert :vfprc_element_value;
waerk :waerk;
kinak :kinak;
koaid :koaid;

}
implemented by method ycl_amdp_prcd_elements=>get_data;
```

Class for CDS

_____

```abap
CLASS ycl_amdp_prcd_elements DEFINITION
PUBLIC
FINAL
CREATE PUBLIC .

PUBLIC SECTION.
INTERFACES if_amdp_marker_hdb.
CLASS-METHODS get_data for table FUNCTION YCDS_PRCD_ELEMENTS.
PROTECTED SECTION.
PRIVATE SECTION.
ENDCLASS.

CLASS ycl_amdp_prcd_elements IMPLEMENTATION.

Method get_data
BY DATABASE FUNCTION FOR HDB LANGUAGE SQLSCRIPT
OPTIONS READ-ONLY
USING PRCD_ELEMENTS.

lt_prcd = APPLY_FILTER( prcd_elements, :Sel_opt );

RETURN

SELECT
Prcd_elements.client,
Prcd_elements.knumv,
Prcd_elements.kposn,
Prcd_elements.kschl,
Prcd_elements.kbetr,
Prcd_elements.kpein,
```

```
Prcd_elements.kmein,
Prcd_elements.knumh,
Prcd_elements.kwert,
Prcd_elements.waerk,
Prcd_elements.kinak,
Prcd_elements.koaid

from prcd_elements
* INNER join prcd_elements p2
* on p2.client = prcd.client and
* p2.knumv =prcd.knumv

Order by Prcd_elements.knumv,
Prcd_elements.kposn;

ENDMethod.

ENDCLASS.
```

——————————-

Called in program as below :

```
SPAN {
font-family: "Courier New";
font-size: 15pt;
color: #000000;
background: #FFFFFF;
}
.L0S31 {
font-style: italic;
color: #808080;
}
```

```
.LOS33 {
color: #4DA619;
}
.LOS52 {
color: #0000FF;
}
.LOS55 {
color: #800080;
}
.LOS70 {
color: #808080;
}
MOVE-CORRESPONDING it_tab[] TO it_knumv[].
IF it_knumv[] IS NOT INITIAL.
DELETE ADJACENT DUPLICATES FROM it_knumv COMPARING knumv.
LOOP AT it_knumv INTO wa_knumv.
s_knumv–sign = 'I'.
s_knumv–option = 'EQ'.
s_knumv–low = wa_knumv–knumv.
APPEND s_knumv.
ENDLOOP.

l_where2 = cl_shdb_seltab=>combine_seltabs(
EXPORTING it_named_seltabs =
VALUE #( ( name = 'KNUMV' dref = REF #( s_knumv[] ) ) )
iv_client_field = 'CLIENT' ).

*PERFORM Fetch_prcd.

SELECT * FROM ycds_prcd_elements( sel_opt = @l_where2 ) INTO TABLE @DATA(it_prcd).
```

Sreehari V Pillai | Post author

December 25, 2019 at 12:09 pm

how length your dynamic where clause ? (l_where2) .

Can you try hard coding a simple where clause which might return few result sets and see if you get timeout still ?

Sree

Find us on

Privacy	Terms of Use
Legal Disclosure	Copyright
Trademark	Cookie Preferences
Newsletter	Support