Technical Articles

Vinita Kasliwal

November 27, 2018 6 minute read

AMDP class and methods and how to use in an extractor

This blog contains details of getting started in using AMDP classes

It contains below Topic areas;

1. Introduction
2. Getting started
   - Creating class implementation and method
3. Syntax for common operations
   - Select based on importing parameter coming in the class

- Select based on an internal table
- Select Unique records
- To filter unique values from an internal table
- Combine select from 2 internal tables
- To add select criteria based on a range
- Using case and end case
- Syntax for using SUM

4. Endnotes

Introduction

As mentioned in the help link

https://help.sap.com/doc/abapdocu_751_index_htm/7.51/en-US/abenamdp_classes.htm

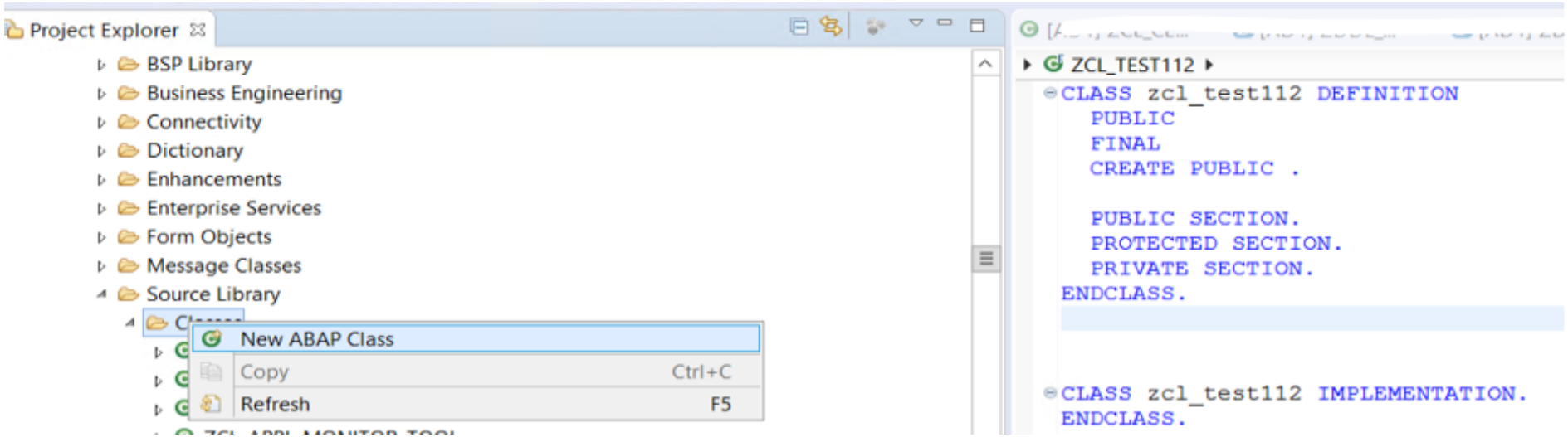
> *An AMDP class is a global class in the class library that contains one or more of the following tag interfaces:*
>
> - *IF_AMDP_MARKER_HDB for the SAP HANA database*
>
> *The names of the interfaces all start with IF_AMDP_MARKER and a suffix indicates the database system for which the ABAP Managed Database Procedures can be implemented in AMDP methods of the class.*
>
> *An AMDP class can contain both regular methods and AMDP methods. It can contain one or more AMDP methods for each database system specified by a tag interface.*

In simple words to select from HANA DB tables instead of using ABAP you can use AMDP which has a slightly different syntax and is way more faster.

Getting started

Creating a class implementation and method.

Create a class and method using HANA studio which automatically populates definition and implementation



Modified the class method to add interface IF_AMDP_MARKER_hdb

Added a Type and table to be returned from the class method get_result and raise exception CX_AMDP_ERROR.

```
CLASS zcl_test_extractor_amdp DEFINITION

PUBLIC

FINAL

CREATE PUBLIC.

PUBLIC SECTION.

INTERFACES if_amdp_marker_hdb .


BEGIN OF ty_attributes,

object_id      TYPE crmt_object_id_db

,guid           TYPE crmt_object_guid

,process_type   TYPE crmt_process_type_db

,P_DESCRIPTION  TYPE crmt_description

,process_desc    type crmt_description

END OF  ty_attributes.


TYPES:

tt_attributes     TYPE STANDARD TABLE OF  ty_attributes,
```

```
CLASS-METHODS get_result

IMPORTING

VALUE(iv_clnt)             TYPE mandt

VALUE(ip_filters)          TYPE string

EXPORTING

VALUE(et_attribute_details) TYPE   tt_attributes

RAISING

cx_amdp_error.

ENDCLASS.
```

Now start with the implementation statement as below

CLASS zcl_test_extractor_amdp IMPLEMENTATION.

and below starts the actual code

Method get_XXX_details by database procedure for hdb language sqlscript
using ekko bseg bkpf.

```
  METHOD get_result
      BY DATABASE PROCEDURE
      FOR HDB
      LANGUAGE SQLSCRIPT
      OPTIONS READ-ONLY
      USING  crmd_orderadm_h    crmc_proc_type     crmc_proc_type_t

   lt_result =

     SELECT DISTINCT

      oh.object_id     AS object_id

     ,oh.process_type  AS process_type

     ,pd.P_DESCRIPTION as P_DESCRIPTION

     ,oh.created_at    AS created_at

     ,oh.created_by    AS created_by

  FROM crmd_orderadm_h

AS oh

  JOIN crmc_proc_type

AS pt ON pt.process_type = oh.process_type and pt.client = IV_CLNT

  JOIN crmc_proc_type_t

AS pd ON pd.process_type = pt.process_type AND pd.langu = 'E'

WHERE oh.object_type = 'BUS2000126' AND oh.process_type = 'ZBKG' and
```

```
oh.client = IV_CLNT

ORDER BY oh.object_id;

 et_attribute_details  = APPLY_FILTER( :lt_result, :ip_filters );

   ENDMETHOD.
```

Syntax for common operations

Select based on importing parameter coming in the class

Here IV_FROMDATE and IV_TODATE are coming as incoming paramaters in the class.

* Fetched the Changed PO details from ECC
lt_ebeln = SELECT ebeln
FROM ekko
WHERE aedat BETWEEN iv_fromdate
AND iv_todate;

Select based on an internal table

Select data based on internal table lt_ebeln.

Note; the internal table is named using a colon:

```
lt_ebeln =   SELECT i.ebeln
   FROM  bkpf as h
   INNER JOIN bseg as i
   on i.bukrs = h.bukrs AND
```

```
        i.belnr = h.belnr AND
        i.gjahr = h.gjahr
    UNION ALL SELECT ebeln from:lt_ebeln;
```

Select Unique records

Select unique records using keyword DISTINCT

lt_re_hyp = SELECT DISTINCT * FROM :lt_re_hyp;

To filter unique values from an internal table

To filter specific value based on value in internal table or variable use Apply_filter.

lt_na = apply_filter(:lt_base, :iv_proj_type_na);

Combine select from 2 internal tables

Get the final result by combining data from multiple internal table

Example lt_cc and lt_wbs using UNION by first selecting from lt_cc and lt_wbs to get the final result in lt_details.

```
    lt_details =
*Cost Center Items
    SELECT DISTINCT
    base_cc.gl_account as gl_account,
    base_cc.ryear as ryear,
    cc_hyp.zz_hpp_project as hp_proj_id,
    from :lt_cc as base_cc
    left outer join :lt_cc_hyp as cc_hyp
```

```
    on base_cc.kostl = cc_hyp.kostl
*WBS Items
    union all
    select distinct
    base_wbs.gl_account as gl_account,
    base_wbs.ryear as ryear,
    wbs_hyp.zz_hpp_project as hp_proj_id,

    from :lt_wbs as base_wbs
    left outer join :lt_wbs_hyp as wbs_hyp
    on base_wbs.projk = wbs_hyp.pspnr ;
```

To add select criteria based on a range

When selecting data use keyword IN and specify the range

```
 LEFT OUTER JOIN tj02t    AS ct ON ct.istat = cs.stat

AND ct.spras = 'E' AND ct.txt04 IN ('CANC','DEF')
```

Using case and end case

CASE and end case to add a specific value

```
et_details =
  select
   base.gl_account as gl_account,
   '' as activity,    case
   when base.period = '001' then 'Jul'
   when base.period = '002' then 'Aug'
```

```
    when base.period = '003' then 'Sep'
    when base.period = '004' then 'Oct'
    when base.period = '005' then 'Nov'
    when base.period = '006' then 'Dec'
    when base.period = '007' then 'Jan'
    when base.period = '008' then 'Feb'
    when base.period = '009' then 'Mar'
    when base.period = '010' then 'Apr'
    when base.period = '011' then 'May'
    when base.period = '012' then 'Jun'
    else 'NA'
    end as period,
    '' as scenario,
    base.year as year,
    SUM ( base.hsl ) as hsl
    from :lt_details as base
    group by gl_account, zzlocbrd, hp_proj_id, prctr, year, period;
```

Syntax for using SUM

To sum up the value using the keyword SUM though I am not still sure if it works as expected

I prefer to get all values and then sum outside of AMDP

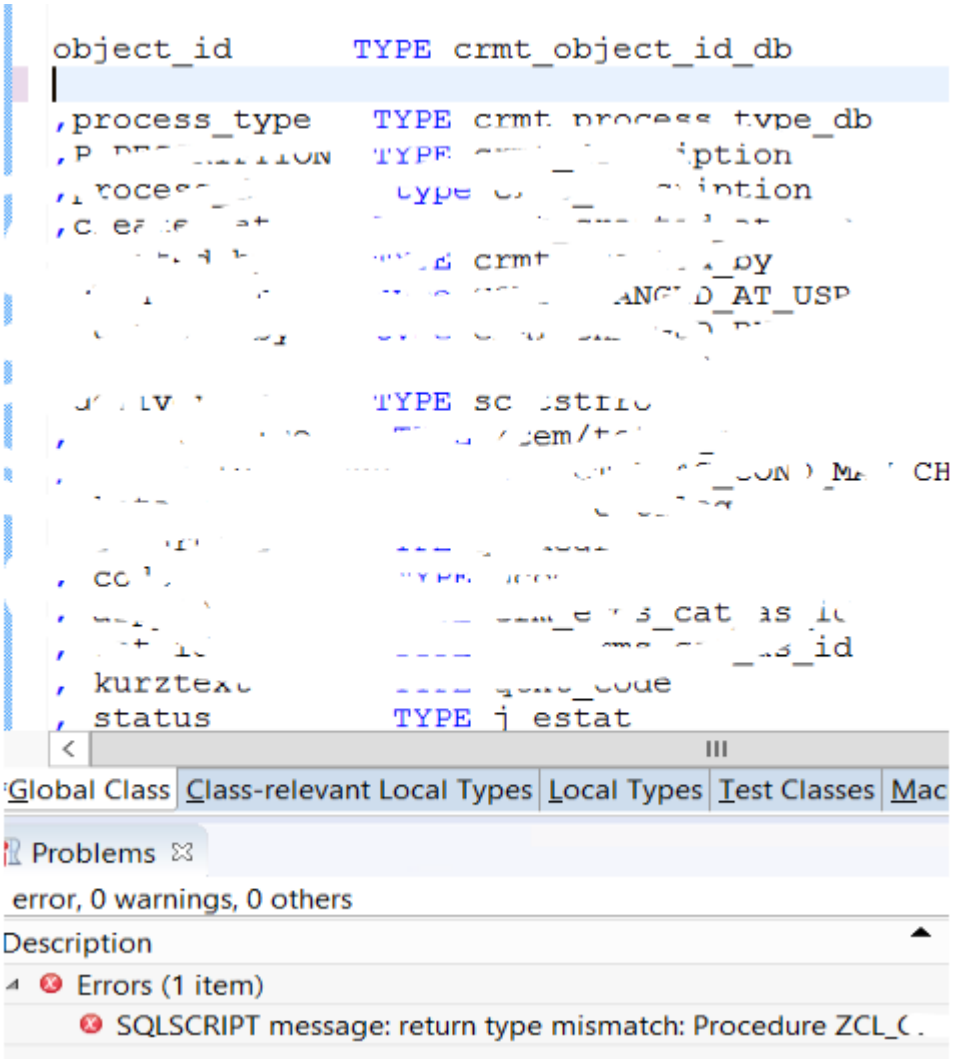
```
lt_sum = SELECT DISTINCT
oh.object_id as object_id ,
sum( pi.net_value ) as total_value

      FROM crmd_orderadm_h     AS oh
```

```
    inner JOIN crmd_orderadm_i     AS oi ON oi.header = oh.guid
  JOIN crmd_pricing_i      AS pi ON pi.guid = oi.guid
  LEFT OUTER JOIN tj02t    AS ct ON ct.istat = cs.stat AND ct.spras = 'E' AND ct.txt04 IN ('CANC','DEF')
  inner JOIN  :et_attribute_details  as base
on oh.object_id = base.object_id
  AND ct.txt30 IS NULL
  GROUP BY oh.object_id , pi.net_value  ;
```

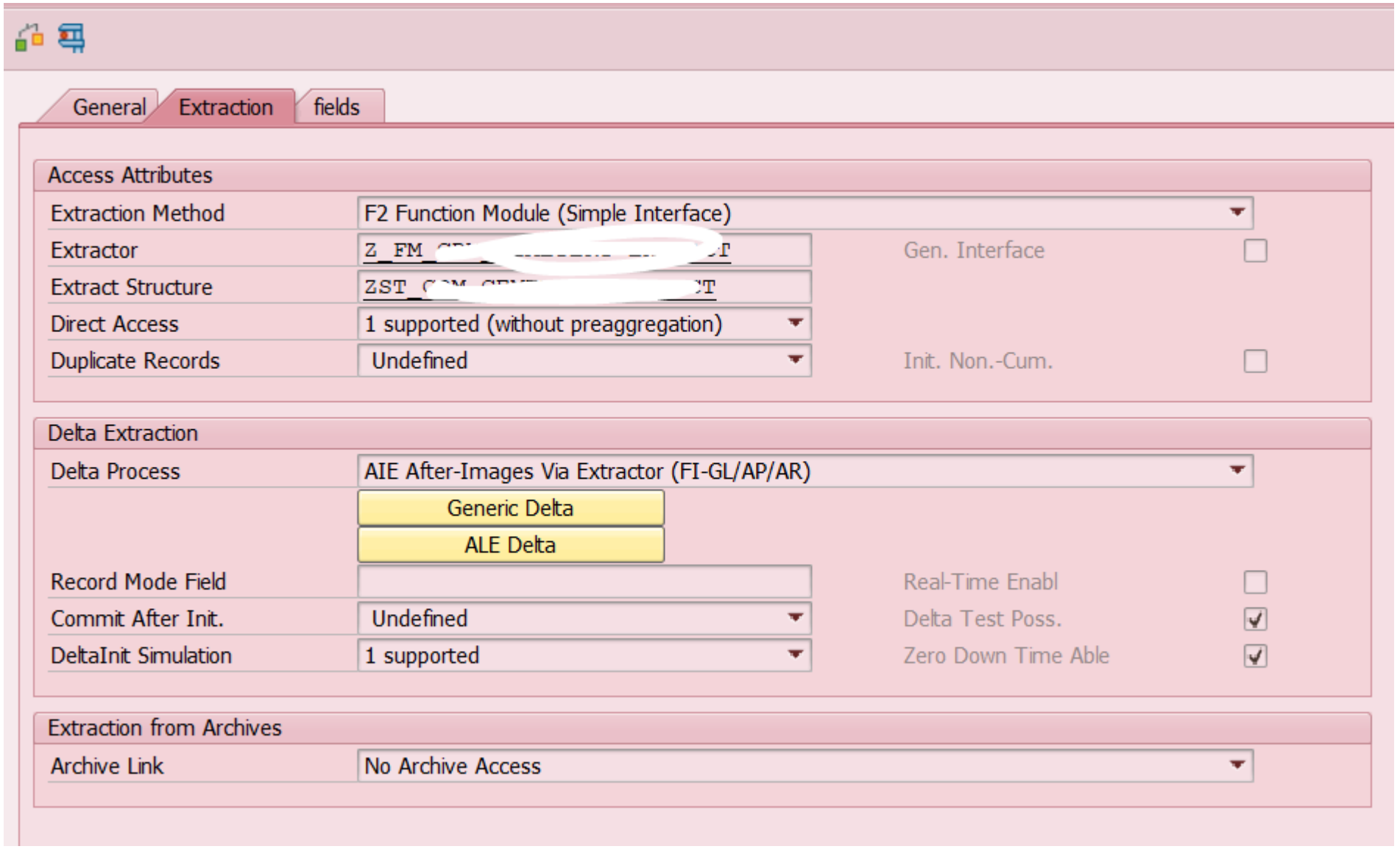
Endnotes

- Important to add IV_MANDT to ensure that specific client data is only selected
- AMDP classes are best called when you want to send data to the BI team and include it as a part of an extractor as they are very fast compared to traditional way of calling a ABAP based select statement
- When you select value it should have same select sequence example if your return table has a sequence as object_id, GUID, Process_type then select statement should also select in the same sequence else it gives an error “SQLSCRIPT Return type mismatch”

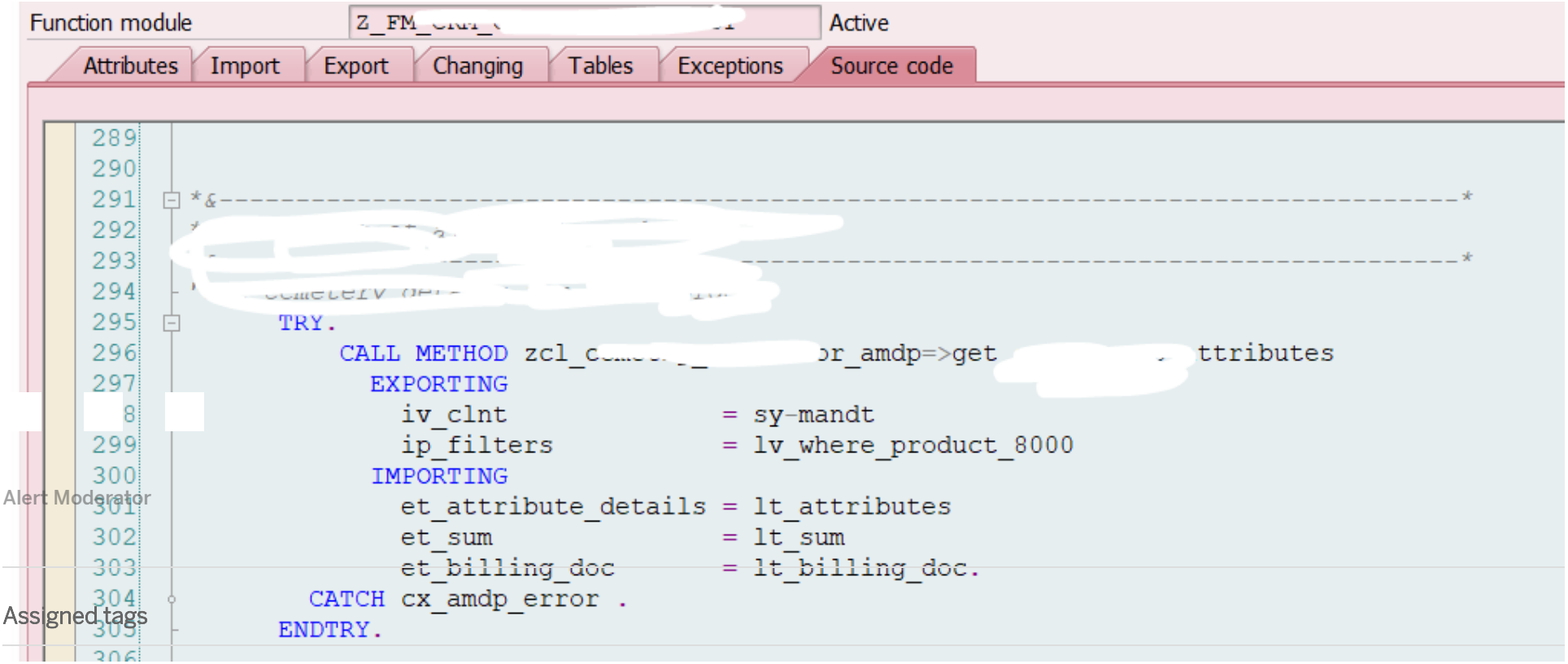


- There is no need to explicitly declare the tables you can mention lt_table = SELECT * .... and it would define the structure to be same as what you select
- AMDP class and methods can only be edited in HANA studio and can only be debugged from there
- Not easy to debug them (or maybe I don't know how to)
- If you require calculations to be done its best to get the data in AMDP and do the remaining calculations outside.

- I found a comparable difference between the speed of result returned from an ABAP report vs AMDP select query AMDP query took 2 min to fetch 10000 records and ABAP query (with same select parameters) took 14 mins
- AMDP class does not allow a lot of calculation to be done and would rather be used just as a select query post calculations, summing calculations etc to be done manually
- Using AMDP in an extractor. Transaction RSA2 for creating an extractor and add Z_ Function module as shown



Inside this FM I am calling the AMDP class



Alert Moderator

Assigned tags

SAP HANA studio | ABAP Development | development tools for SAP Cloud Platform | SAP HANA | SAP HANA Enterprise Cloud |

View more...

Related Blog Posts

I hope you find this blog useful for getting started in using AMDP.

Pl share your comments and feedback

Calling HANA Stored Procedure from Abap class and scheduling it through Process chain (using AMDP – Abap managed database procedure)

By **Former Member** , May 29, 2017

Create a value help (for input parameter / variable) in HANA Calculation view

By **Alvaro Otero** , Dec 05, 2017

Tips – SAP HANA Auto-Documentation – Excel Way

By **Murali Balreddy** , Apr 17, 2018

Related Questions

How to call AMDP method in other AMDP method

By **Former Member** , Dec 19, 2017

AMDP Error: Invalid function or procedure

By **sudhanshu sharma** , Mar 07, 2017

How to use select options for AMDP

By **RAMESH SAHOO** , May 01, 2018

4 Comments

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

Michael Wegener

November 27, 2018 at 5:08 am

Thanks for this interesting post.

I have a question regarding section 'Select based on an internal table': Is there a specific reason to use the UNION approach for this? I would have expected an INNER JOIN or a WHERE ... IN (SELECT ...) with the internal table.

Also it's worth noticing that you can use the ABAP class 'CL_SHDB_SELTAB' to create the filter string based on ABAP SELECT-OPTIONS, as described in this blog post https://blogs.sap.com/2015/03/30/handling-of-select-options-parameters-within-amdp/.

Best regards, Michael

Like (1)

Michelle Crapo

November 27, 2018 at 1:58 pm

Thank you for the comment – not my blog, but reading this makes it a richer experience. So thank you.

Like (0)

Vinita Kasliwal | Post author

December 15, 2018 at 5:43 pm

Oh ok noted. I think there are different ways of doing it mine was just one of them

Like (0)

Michelle Crapo

November 27, 2018 at 1:59 pm

Nice blog – the step by step is very helpful.

Like (1)

Share & Follow