SAP BW - PSA/Change Log Deletion Governance

Applies to:

SAP Net Weaver 2004s BI 7.0 Ehp1 SP 05. For more information, visit EDW homepage

Summary

This article suggests importance of PSA/Change log deletion process and governance around it. In any BW database one third space is being occupied by the PSA/Change Log. A proper strategy should be there with business involvement for PSA/Change Log Deletion and expectations set accordingly.

Author: Pradeep Banwar

Company: Infosys Technologies Limited

Created on: 15th October 2010

Author Bio

The author has been associated with Infosys Technologies Limited for 3 years and has been in a Sr, consultant role in the Process and Domain consulting Stream. He has been involved in a BW upgrade and is currently placed as the Platform lead of the complex Enterprise Data Warehouse Platform of a renowned Client in this domain.

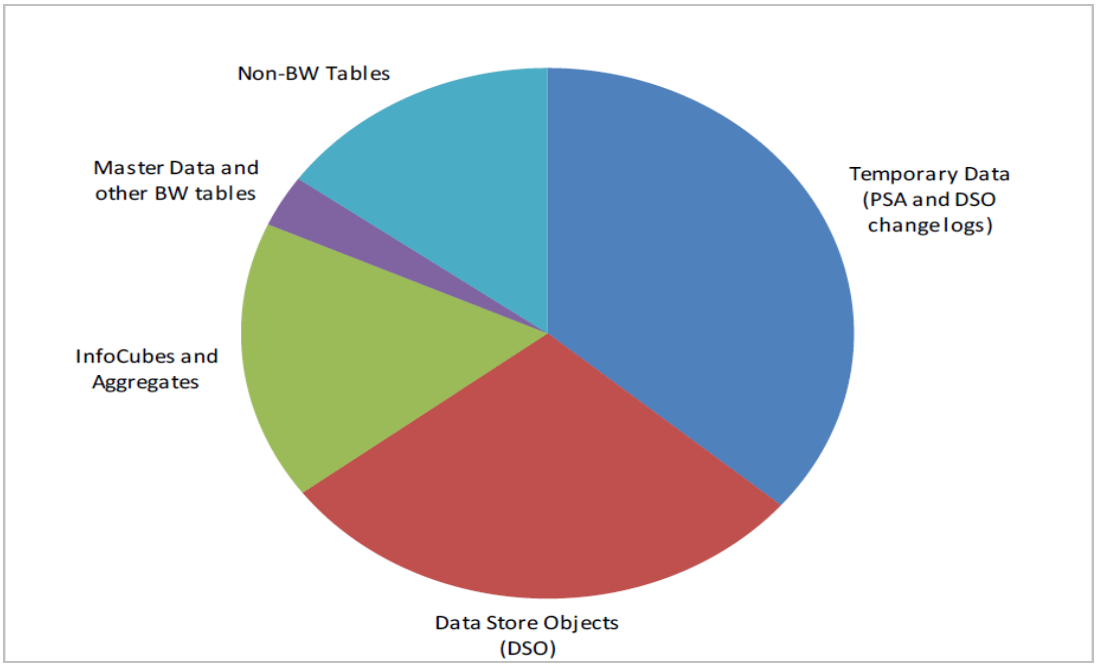
Table of Contents

Introduction

Even the most modern and technologically advanced database systems can suffer from performance bottlenecks caused by large volumes of data. In applications, this can lead to poor system performance, while from an administration point of view, it can cause increases in the use of resources. High volumes of data can also have a considerable effect on the total cost of ownership of a system, despite falling storage prices.

According to SAP Data Volume Strategy, data deletion is also one of the methodologies used to check potential size and growth reductions.

In any BW database One third space is being occupied by temporary data (PSA and DSO Change Log) as shown below.



Importance of PSA/Change Log

Persistent Staging Area stores data in the original format while being imported from the source system, PSA allows for quality checks before the data are loaded into their destinations such as ODS objects or Info cubes

Persistent Staging area is used as a temporary space to hold data before it is loaded to the SAP BI system. As the operational system OLTP cannot be kept out of service for long, we use the PSA. All the data in the source system is copied as it is to the PSA. Transformations are then carried out on PSA data before it gets loaded in the SAP BI system.

If we set the PSA when we are extracting data, you get improved performance. The temporary storage facility in the PSA also allows to check and change the data before the update into data targets. In contrast to a data request with IDocs, a data request in the PSA also gives various options for a further update of the data in the data targets. The possible coupling of the loading process from the further processing in BW likewise contributes to an improved loading performance. The operative system is not debited if data errors first appear with further processing.

A persistent staging area is the foundation that provides this functionality. When bugs are found in the load process or the business wants to change the rules, we can design a systematic approach to removing, reloading and tracking data loads as it propagates throughout the warehouse.

Audience & Recommended Use

The entire BW community specially the Production Support team. The Service Delivery team and Platform Team also needs to be aware of the situation. The Stake holders involved during the implementation and later during maintenance.

Need for Governance

- PSA Occupies one third of the BW Database space.
- It is very important that PSA's should be deleted periodically.
- PSA/Change log deletion should be scheduled via process chains when data loads are not running and user activity is minimal.
- The PSA deletion Strategy should be reviewed every year.
- After successful completion of the project it is often forgotten to schedule PSA/Change log data deletion jobs
- Strict Governance should be in place so that PSA deletion in included for the new projects going live.
- Improves overall performance of the system

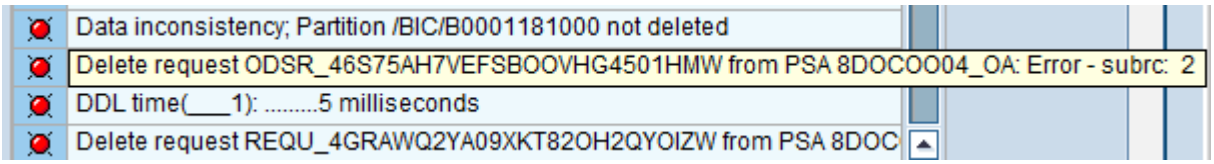
Importance of PSA/Change Log Deletion. In any BW database One third space is being occupied by the PSA/Change Log. A proper strategy should be there with business involvement for PSA/Change Log Deletion and expectations set accordingly.

Strategy should be to define retention times for PSA records for all Data Sources as well as for DSO change logs and delete outdated data. Schedule regular deletion accordingly.

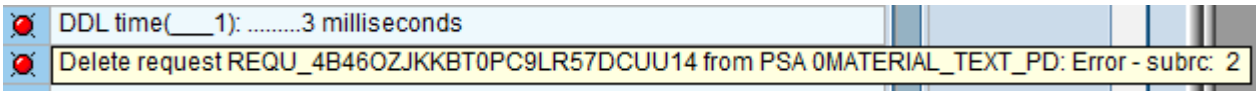
Challenges

- With the new functionality of EhP1 we need to give the DSO name in the change log deletion variant which is being done by F4 functionality. All the DSO's are not getting captured at one go and there are chances when the Change logs are not getting deleted.
- In big landscapes the PSA/Change log count are in thousands and it is impossible to find out which PSA's/Change logs are not getting deleted. ST14 transaction also shows the top xx PSA & Change Log tables but this is internal SAP transaction and not is use always.
- A few Change log and PSA variant are not getting executed successfully and below error appears.

Change Log



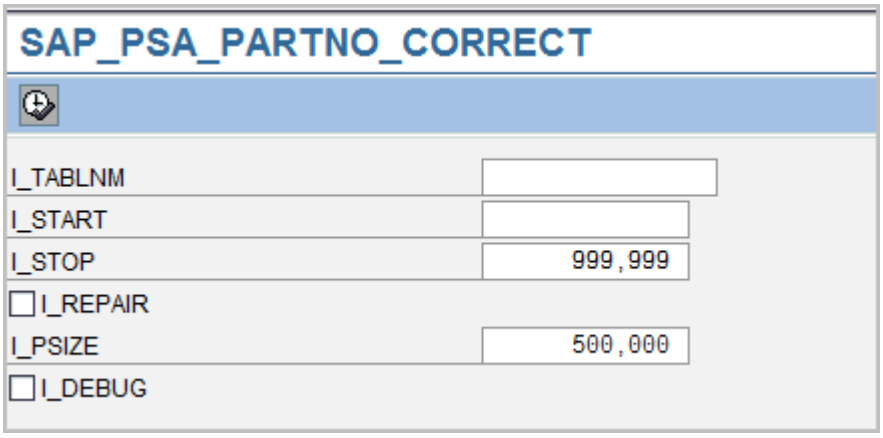
PSA



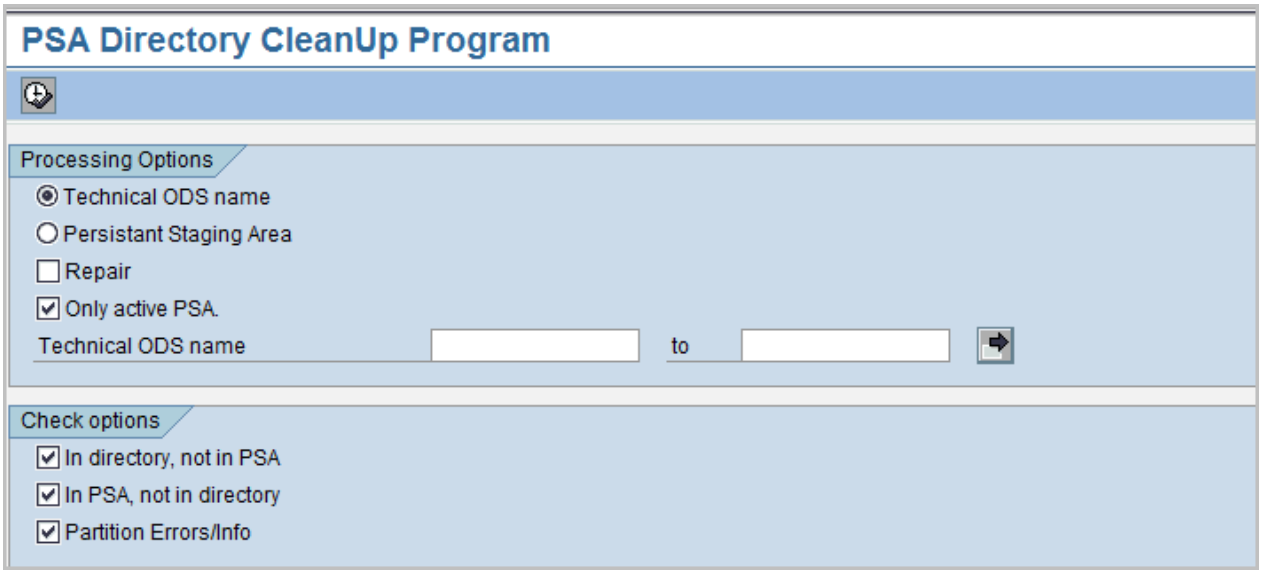
Steps to Overcome Challenges

Step 1: PSA/Change log Process chain failing for few variants.

- Identify the PSA's which are not getting deleted inspite of repeated running of process chains.
- Check whether SAP Note 1063105 (PSA Clean up Directory) is implemented in the BW system or not.
- Then execute the report **SAP_PSA_PARTNO_CORRECT** for the concerned PSA first in check and then in repair mode as stated in note 1051664 as shown below.



- Then again execute report **rsar_psa_cleanup_directory** first in check mode for the concerned PSA and then in repair mode as shown below.



Step 2: PSA/Change Log not scheduled for deletion

Take help of a custom built ABAP program which gives out as the list of PSA's which have huge requests. It helps find out the temporary tables (PSA & Change Log) which are not getting deleted. It is particularly very helpful for big landscapes involving many temporary tables (PSAs & Change Logs) where this check would ensure proper house-keeping for the fast growing tables.

Step 3: Details of Custom built ABAP Program

Function is to see which PSA is not being deleted periodically and to see which PSA table is occupying significant space.

REPORT : ZBW_IDENTIFY_PSA

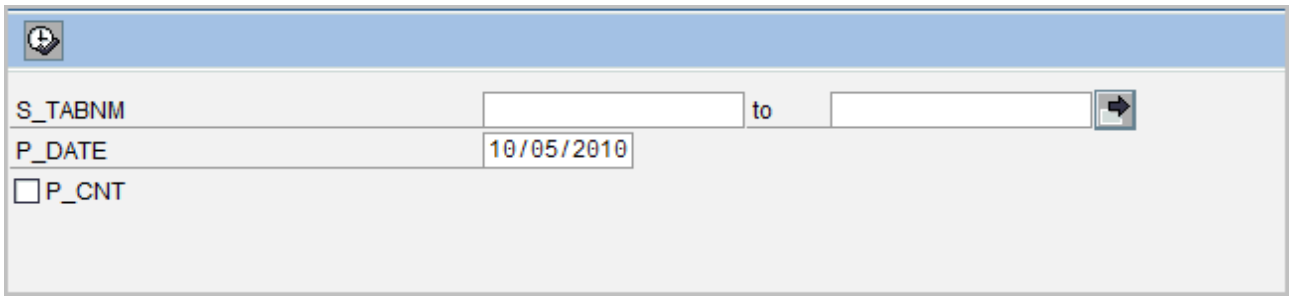
```
TABLES: rstsods.
TYPE-POOLS: slis.

TYPES: BEGIN OF ty_display,
Tabname   TYPE rstsods-odsname,
count           TYPE rscewcount,
odsname(20)     TYPE c,
records         TYPE nrows,
END       OF ty_display,

BEGIN OF ty_rsreqicods,
tabname   TYPE rsreqicods-tabname,
timestamp       TYPE rsreqicods-timestamp,
req_date  TYPE sy-datum,
END       OF ty_rsreqicods,

BEGIN OF ty_rstsods,
odsname   TYPE rstsods-odsname,
odstech   TYPE rstsods-odsname_tech,
dateto T  YPE rstsods-dateto,
END       OF ty_rstsods.

DATA:
 wa_display     TYPE ty_display,
wa_rsreqicods   TYPE ty_rsreqicods,
wa_rstsods      TYPE ty_rstsods,
 t_display      TYPE STANDARD TABLE OF ty_display INITIAL SIZE 0,
 t_rsreqicods   TYPE STANDARD TABLE OF ty_rsreqicods INITIAL SIZE 0,
 t_rstsods      TYPE STANDARDTABLE OF ty_rstsods INITIAL SIZE 0.


SELECT-OPTIONS:
s_tabnm FOR rstsods-odsname.
PARAMETERS:
p_date          LIKE sy-datum OBLIGATORY DEFAULT sy-datum,
p_cnt           TYPE c AS CHECKBOX.

START-OF-SELECTION.
  DATA:
lc_timestamp(14)       TYPE c,
l_timestamp            TYPE rsreqicods-timestamp.
  CONCATENATE p_date '000000' INTO lc_timestamp.
```

```
l_timestamp =  lc_timestamp.

  SELECT tabname
 timestamp
INTO TABLE t_rsreqicods
FROM rsreqicods
WHERE timestamp < l_timestamp
AND typ = 'O' AND tabname IN s_tabnm.

  IF sy-subrc <> 0.
    WRITE: / 'No matching records found.'(001).
    EXIT.
  ENDIF.

  LOOP AT t_rsreqicods INTO wa_rsreqicods.
    lc_timestamp = wa_rsreqicods-timestamp.
    wa_rsreqicods-req_date = lc_timestamp(8).
    MODIFY t_rsreqicods FROM wa_rsreqicods TRANSPORTING req_date.
  ENDLOOP.

  SELECT odsname
odsname_tech
dateto
INTO TABLE t_rstsods
 FROM rstsods
  WHERE odsname IN ( SELECT DISTINCT tabname FROM rsreqicods WHERE
  timestamp < l_timestamp AND typ = 'O' AND tabname IN s_tabnm ).
  SORT
t_rstsods BY odsname dateto.

  LOOP AT t_rsreqicods INTO wa_rsreqicods.
    CLEAR wa_display.
wa_display-tabname = wa_rsreqicods-tabname.
    wa_display-count = 1.

READ TABLE t_rstsods WITH KEY odsname =  wa_rsreqicods-tabname
TRANSPORTING NO FIELDS BINARY SEARCH.

    IF sy-subrc = 0.
      LOOP AT t_rstsods INTO wa_rstsods FROM sy-tabix.
        IF wa_rstsods-odsname <> wa_rsreqicods-tabname.
          EXIT.
        ENDIF.

        IF
wa_rstsods-dateto >= wa_rsreqicods-req_date.
          wa_display-odsname = wa_rstsods-odstech.
          EXIT.
        ENDIF.
      ENDLOOP.
    ENDIF.

    COLLECT wa_display INTO t_display.
  ENDLOOP.
  LOOP AT t_display INTO wa_display.
```

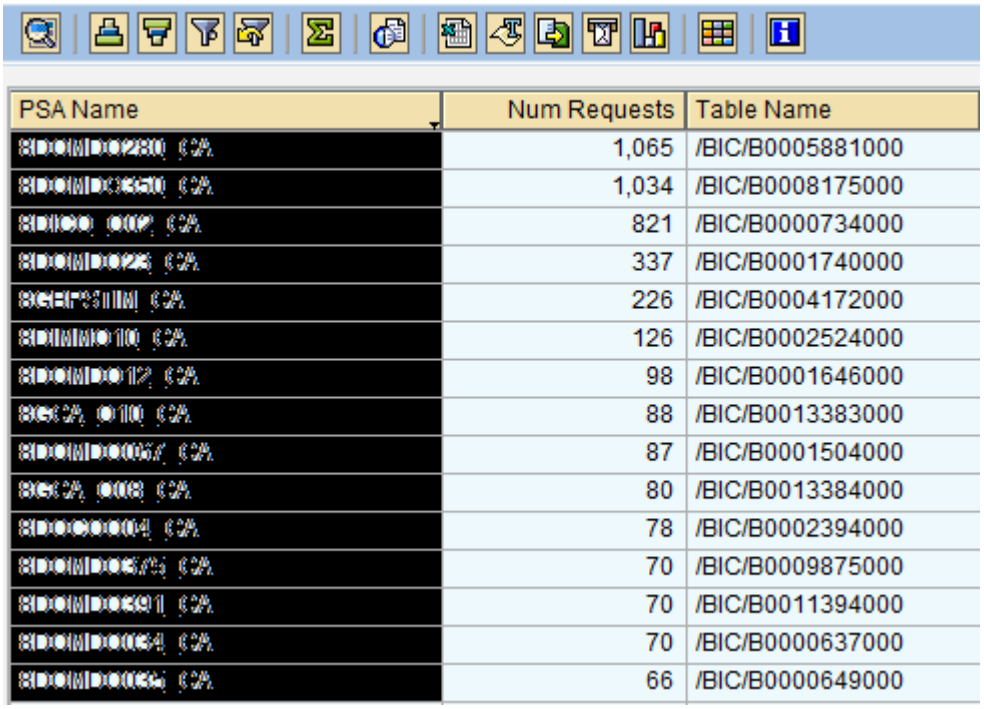
```
    IF wa_display-odsname <> ''.


      SELECT tabname
INTO wa_display-odsname
FROM dd02l
UP TO 1 ROWS
WHERE       tabname = wa_display-odsname.
      ENDSELECT.

      IF sy-subrc = 0.
        IF p_cnt = 'X'
.
          SELECT COUNT(*)
INTO wa_display-records
FROM (wa_display-odsname).
        ENDIF.
      ELSE.

        CLEAR wa_display-records.
CONCATENATE '(' wa_display-odsname ')'
 INTO  wa_display-odsname
SEPARATED BY space.
      ENDIF.

      MODIFY t_display FROM wa_display.
    ENDIF.
  ENDLOOP.

  IF p_cnt = 'X'.
    SORT t_display BY records       DESCENDING.
  ELSE.
    SORT t_display BY count DESCENDING.
  ENDIF.

END-OF-SELECTION.

  DATA:
wa_fc     TYPE slis_fieldcat_alv,
t_fc      TYPE STANDARD TABLE OF slis_fieldcat_alv INITIAL SIZE 0.

  wa_fc-tabname = 'T_DISPLAY'.

  wa_fc-col_pos = 1.
wa_fc-fieldname = 'TABNAME'.
wa_fc-seltext_s = wa_fc-seltext_m = 'PSA Name'(005).
  wa_fc-outputlen = 30.
APPEND wa_fc TO t_fc.

  wa_fc-col_pos = 2.
 wa_fc-fieldname = 'COUNT'.
wa_fc-seltext_s =wa_fc-seltext_m = 'Num Requests'(006).
  wa_fc-outputlen = 12. APPEND
wa_fc TO t_fc.
  wa_fc-col_pos = 3.
```

```
wa_fc-fieldname = 'ODSNAME'.
 wa_fc-seltext_s =  wa_fc-seltext_m = 'Table Name'(007).
  wa_fc-outputlen = 20.
APPEND wa_fc TO t_fc.

  IF p_cnt = 'X'.
    wa_fc-col_pos = 4.
 wa_fc-fieldname = 'RECORDS'.
    wa_fc-seltext_s = wa_fc-seltext_m = 'Record Count'(008).
    wa_fc-outputlen = 20.
APPEND wa_fc TO t_fc.
  ENDIF.

  CALL FUNCTION    'REUSE_ALV_GRID_DISPLAY'
    EXPORTING
      it_fieldcat   = t_fc[]
    TABLES
      t_outtab      = t_display[]
    EXCEPTIONS
      program_error = 1
      OTHERS        = 2.
  IF sy-subrc <> 0.
    WRITE: / 'ALV Display Error:'(003), sy-subrc.
  ENDIF.
```

The selection screen will be as follows.



The output will be as below.



PSA Name	Num Requests	Table Name
[illegible]	1,065	/BIC/B0005881000
[illegible]	1,034	/BIC/B0008175000
[illegible]	821	/BIC/B0000734000
[illegible]	337	/BIC/B0001740000
[illegible]	226	/BIC/B0004172000
[illegible]	126	/BIC/B0002524000
[illegible]	98	/BIC/B0001646000
[illegible]	88	/BIC/B0013383000
[illegible]	87	/BIC/B0001504000
[illegible]	80	/BIC/B0013384000
[illegible]	78	/BIC/B0002394000
[illegible]	70	/BIC/B0009875000
[illegible]	70	/BIC/B0011394000
[illegible]	70	/BIC/B0000637000
[illegible]	66	/BIC/B0000649000

This program identifies both PSA and Change Log tables which are not getting deleted by seeing at the number of requests.

References

https://www.sdn.sap.com/irj/scn/weblogs?blog=/pub/wlg/5801

Deleting the Requests from the PSA and Change Log Tables in Business Intelligence

http://www.ittestpapers.com/blogs/sap-bw--psa-and-changelog-deletion.html

For more information, visit the Business Intelligence homepage.

For more information, visit 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.