Deletion of Old Process Chain Execution Logs in BW/BI

Applies to:

SAP BW 3.x and higher versions as BW system. For more information, visit the [Business Intelligence homepage](#).

Summary

This document is intended to show how to delete old process chain execution logs in SAP BW system using ABAP program in order to clear database space occupied by not required old process chain execution logs.

Author: Amol Ishwarlal Jaiswal

Company: Infosys Technologies Ltd.

Created on: 20 March 2010

Author Bio

Amol Jaiswal is a BI consultant with over 2.5 years of industry experience in implementation and maintenance of SAP BW/BI systems. Presently, Amol is working with Infosys Technologies Ltd and is involved in execution of SAP BW/BI projects.

Table of Contents

Introduction to Process Chains

A **Process Chain** is a sequence of processes those are scheduled to wait in the background for an event. Some of these processes trigger a separate event that can, in turn, start other processes. A **Process** is a procedure inside of or external to a SAP system with a defined beginning and end. A Process is characterized by

- Process type   : The kind of process.

- Process variant : Name of the process and defines its characteristics.

- Process instance: Messages and instances for its successors.

At runtime the Process chain execution is as follows.

- Get the variant and the predecessor list.

- Instantiate the process object.

- Ask predecessors for information.

- Execution of the defined process.

- Report ending with status and instance.

Every Process chain is started using a Start Process, that cannot have any predecessor processes and it cannot be successor of any other processes. Only one start process can be used per process chain.

In-order to automate the runs in BW the processes are brought together in Process Chains. Process chains can be used as a scheduling and monitoring tool. The tasks such as data loads, Reporting Agent jobs, or index rebuilds can be accomplished using process chains.

The following transaction codes can be helpful while dealing with the process chains

RSPC   – Maintenance of process chains.

RSPCM – Process chain Monitor.

After execution of process chain, the logs are created which are stored in below mentioned database tables in the system.

RSPCPROCESSLOG: Logs for the Chain Runs

RSPCLOGCHAIN: Cross-Table Log ID / Chain ID

The older process chain logs are referred rarely and hence unnecessarily occupy the database space. The program which we will be creating will delete these unnecessary log entries and keeping only those which are useful.

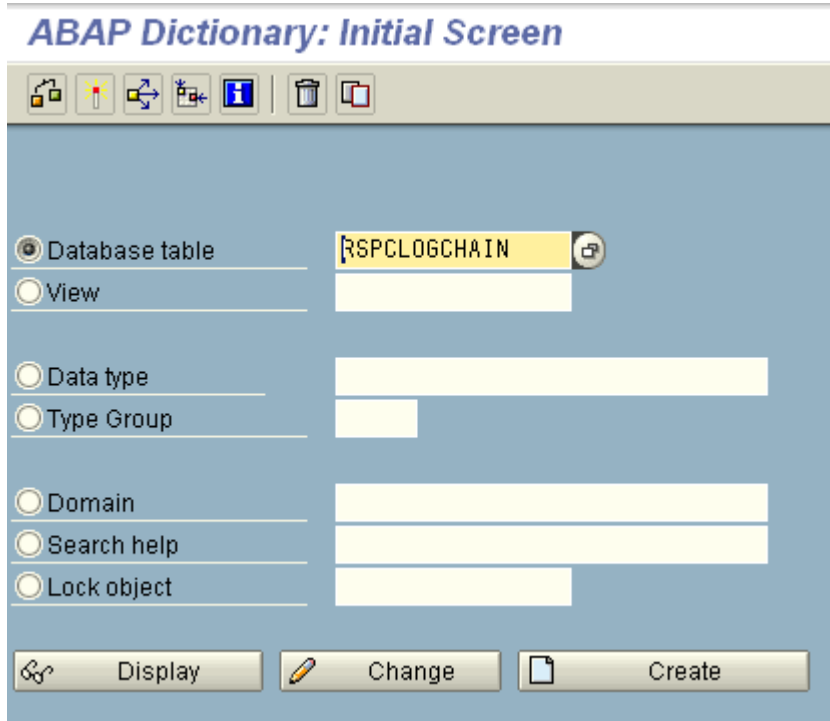To check these tables in the system you can use transaction code SE11.

ABAP Dictionary: Initial Screen

⊙ Database table   `RSPCLOGCHAIN` 　
○ View

○ Data type
○ Type Group

○ Domain
○ Search help
○ Lock object

👓   Display   ✏   Change   🗋   Create

Fig-1: SE11 for RSPCLOGCHAIN

Dictionary: Display Table

Transp. table   RSPCLOGCHAIN   Active
Short Text   Cross-Table Log ID / Chain ID

Attributes | Delivery and Maintenance | Fields | Entry help/check | Currency/Quantity Fields

1 / 18

Field	Key	Initi	Data element	Data Ty	Length	Decim	Short Text	Group
CHAIN_ID	✓	✓	RSPC_CHAIN	CHAR	25	0	Process Chain	
LOG_ID	✓	✓	RSPC_LOGID	CHAR	25	0	Log-ID of a Process Chain Run	
DRAW_TYPE			RSPC_TYPE	CHAR	10	0	Process Type	
DRAW_VARIANT			RSPC_VARIANT	CHAR	30	0	Process Variant (Name)	
DATUM			SYDATUM	DATS	8	0	Date and Time, Current (Application Server) Date	
ZEIT			SYUZEIT	TIMS	6	0	Date and Time, Current Application Server Time	
ANALYZED			RS_BOOL	CHAR	1	0	Boolean	
META_LOG			RSPC_LOGID	CHAR	25	0	Log-ID of a Process Chain Run	
CONS_REPORTED		✓	RSTIMESTMPL	DEC	21	7	UTC time stamp in long form (YYYYMMDDhhmmss,mmmuuun)	
META_DEST		✓	RFCDEST	CHAR	32	0	Logical Destination (Specified in Function Call)	
META_WIID		✓	SWW_WIID	NUMC	12	0	Work item ID	
META_LOGSYS		✓	LOGSYS	CHAR	10	0	Logical system	
META_REPORTED		✓	RSTIMESTMP	DEC	15	0	UTC Time Stamp in Short Form (YYYYMMDDhhmmss)	
SYNCHRONOUS		✓	RSPC_SYNCHRONOUS	CHAR	1	0	Synchronization of Processes	
META_API		✓	RS_BOOL	CHAR	1	0	Boolean	
REPORTED_STATUS		✓	RSPC_STATE	CHAR	1	0	Status of Process	
SYNC_DEBUG		✓	RSPC_SYNCHRONOUS	CHAR	1	0	Synchronization of Processes	

Fig-2: Fields of table RSPCLOGCHAIN

As shown in *fig-2* the fields **CHAIN_ID** and **LOG_ID** form the primary key of the table RSPCLOGCHAIN. The field LOG_ID stores the system generated unique ID for the particular execution of process chain and this field is the link between the tables RSPCLOGCHAIN and RSPCPROCESSLOG. The field CHAIN_ID stores the technical name of the executed process chain. The fields **DATUM** and **ZEIT** store the date and time of the execution of process chain respectively. We will be using all these fields in our program in order to delete old process chain execution logs.

Fig-3: SE11 for RSPCPROCESSLOG

Fig-4: Fields of table RSPCPROCESSLOG

Now we will start creating this program for deletion of old process chain execution logs.

Process Chain Log Deletion Program

For creation of the ABAP program transaction code SE38 is used.
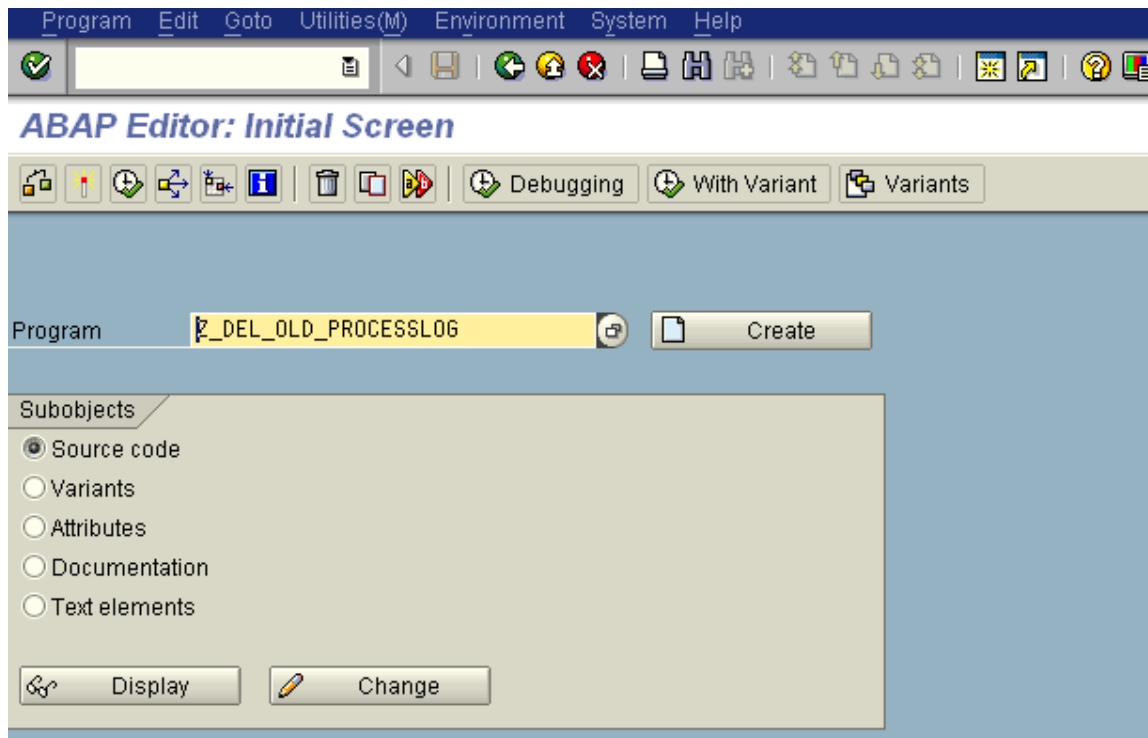
Fig-5: SE38 for Process chain log Deletion Program

Enter the program name as Z_DEL_OLD_PROCESSLOG in the input field as shown in *fig-5* and click on the button Create, a pop up will come for input of description and type of the program as shown in *fig-6*. Please enter the details as shown below and click on Save button.

Title	Deletion of Old Process Chain Execution Logs	
Original language	EN English	
Created	18.01.2009	JAISWAM2
Last changed by		
Status	New(Revised)	

Attributes

Type	Executable program
Status	Customer production program
Application	Business Information Warehouse
Authorization Group	
Logical database	
Selection screen	
☐ Editor lock	☑ Fixed point arithmetic
☑ Unicode checks active	☐ Start using variant

✔ Save

Fig-6: Attributes of Program Z_DEL_OLD_PROCESSLOG

One more pop up will come after clicking on Save button and will ask for package, you can select it as local object and save. Then you will be directed to ABAP editor screen.

Please enter below mentioned ABAP code to the editor and activate the program. There is functionality provided by program to delete the process chain execution logs depending on the execution date of the process chain. If the date selection is not provided by user on the selection screen, only the latest process chain execution log will remain in the database tables and rest all will be deleted. For more explanations of the statements in the program, please refer to the comments in the blue color.

```
*&---------------------------------------------------------------------*
* The purpose of this program is to keep only useful process chain logs*
* and delete other logs in tables RSPCLOGCHAIN and RSPCPROCESSLOG as   *
* some process chain logs are not required any more and they           *
* unnecessarily occupy database space.                                 *
*&---------------------------------------------------------------------*
REPORT  Z_DEL_OLD_PROCESSLOG.
*---------------------------------------------------------------------*
*          Start of Program
*---------------------------------------------------------------------*
*---------------------------------------------------------------------*
*          DB-Tables
*---------------------------------------------------------------------*
TABLES: RSPCLOGCHAIN,    " Cross-Table Log ID / Chain ID
        RSPCPROCESSLOG. " Logs for the Chain Runs
*---------------------------------------------------------------------*
*          Variables/Internal Tables
*---------------------------------------------------------------------*
DATA: COUNT1 TYPE P,
      COUNT2 LIKE COUNT1.
DATA : LV_RC LIKE SY-SUBRC.
DATA: tb_chain LIKE RSPCLOGCHAIN OCCURS 0 WITH HEADER LINE.
*---------------------------------------------------------------------*
*          Selection-Screen
*---------------------------------------------------------------------*
SELECTION-SCREEN BEGIN OF BLOCK b01 WITH FRAME TITLE text-001.
SELECTION-SCREEN BEGIN OF LINE.
SELECTION-SCREEN COMMENT 1(31) text-002 FOR FIELD pa_lat.
PARAMETERS: pa_lat RADIOBUTTON GROUP rad1 USER-COMMAND CLICK DEFAULT 'X'.
SELECTION-SCREEN END OF LINE.
SELECTION-SCREEN SKIP.
SELECTION-SCREEN BEGIN OF LINE.
SELECTION-SCREEN COMMENT 1(31) text-003 FOR FIELD pa_dat.
PARAMETERS: pa_dat RADIOBUTTON GROUP rad1.
SELECTION-SCREEN END OF LINE.
SELECTION-SCREEN BEGIN OF BLOCK b02 WITH FRAME TITLE text-004.
SELECT-OPTIONS: so_date FOR RSPCLOGCHAIN-DATUM MODIF ID ABC OBLIGATORY.
SELECTION-SCREEN END OF BLOCK b02.
SELECTION-SCREEN END OF BLOCK b01.
*---------------------------------------------------------------------*
*          At Selection-Screen
*---------------------------------------------------------------------*
AT SELECTION-SCREEN OUTPUT.
  LOOP AT SCREEN.
    IF SCREEN-GROUP1 = 'ABC'.
      IF pa_dat = 'X'.
        screen-active = 1.
        modify screen.
      ELSE.
        screen-active = 0.
        modify screen.
      ENDIF.
    ENDIF.
  ENDLOOP.
*---------------------------------------------------------------------*
```

```
*           Start-Of-Selection
*---------------------------------------------------------------------*
START-OF-SELECTION.
*take complete data from RSPCLOGCHAIN into internal table tb_chain.
  select * from RSPCLOGCHAIN into table tb_chain.
*sort tb_chain in descending order according to chain_id, datum, zeit
* so that latest record for particular chain id should be at top.
  SORT tb_chain descending by CHAIN_ID DATUM ZEIT.
  clear tb_chain.
  LOOP AT tb_chain.
*delete latest record from internal table so that it should contain only
*those records which we need to delete from both tables.
    ON CHANGE OF tb_chain-CHAIN_ID.
* keep latest records only
      IF pa_lat = 'X'.
        delete tb_chain.
* delete only those records which meet date range selection option
      ELSEIF pa_dat = 'X'.
        delete tb_chain where CHAIN_ID = tb_chain-CHAIN_ID
                        and   DATUM NOT IN so_date.
      ENDIF.
    ENDON.
  ENDLOOP.
  LOOP AT tb_chain.
*delete RSPCPROCESSLOG records where log_id is matching with that of tb_chain.
    DELETE FROM RSPCPROCESSLOG WHERE LOG_ID = tb_chain-log_id.
*counter for taking note of how many records are deleted
    count1 = count1 + SY-DBCNT.
    CLEAR SY-DBCNT.
*delete RSPCLOGCHAIN records where log_id is matching with that of tb_chain.
    DELETE FROM RSPCLOGCHAIN WHERE LOG_ID = tb_chain-log_id.
*counter for taking note of how many records are deleted
    count2 = count2 + SY-DBCNT.
    CLEAR SY-DBCNT.
  ENDLOOP.
  WRITE: / 'The no. of records deleted from RSPCPROCESSLOG  :', count1,
         / 'The no. of records deleted from RSPCLOGCHAIN:', count2.
*---------------------------------------------------------------------*
*           End of Program
*---------------------------------------------------------------------*
```

The text symbols and Selection texts for selection screen should be maintained as well. This can be done as shown in *fig-7*.

Fig-7: Text Elements for program Z_DEL_OLD_PROCESSLOG

Then you can enter the description you want for these Text symbols and Selection Texts as shown in *fig-8* and *fig-9*.

Fig-8: Text Symbols for program Z_DEL_OLD_PROCESSLOG

Fig-9: Selection Texts for program Z_DEL_OLD_PROCESSLOG

Then activate this as well and go back to ABAP editor and again activate our program and click on Direct Processing button or press F8. This will take you to the selection screen of the program as shown in *fig-10*.
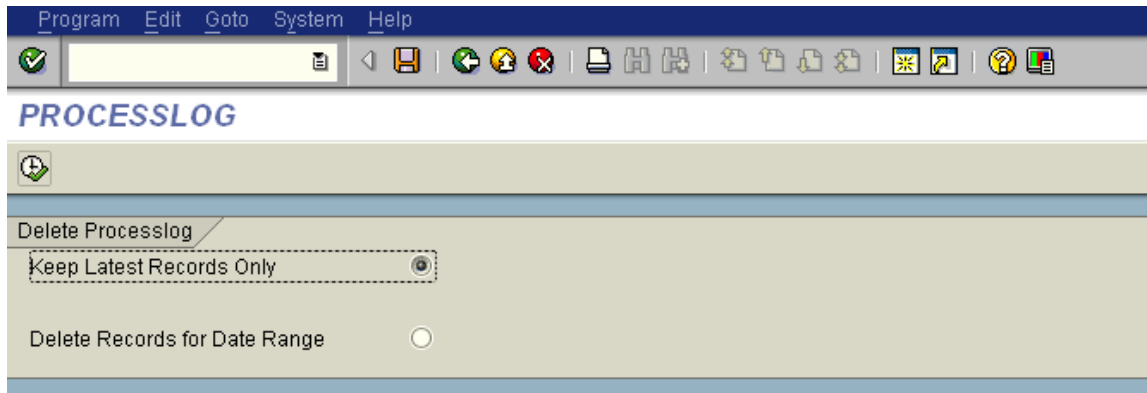
Fig-10: Selection Screen of program Z_DEL_OLD_PROCESSLOG

The radio button 'Keep Latest Records Only' will be selected by default as shown in *fig-10*. If user clicks on execute button or press F8 button, all the old process chain execution logs for each process chain will be deleted except the one with latest date and time.

If user selects radio button 'Delete Records for Date Range', a mandatory input field for date range will appear on the screen as shown in *fig-11*. Let's delete the process chain logs older than 1st June, 2008.
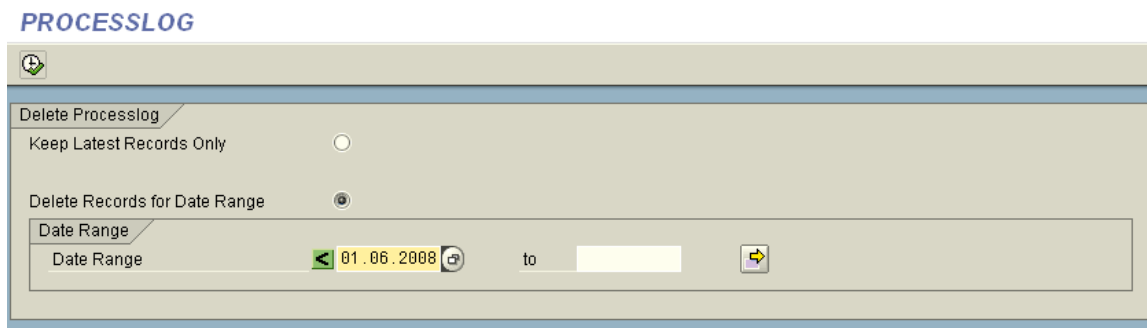
Fig-11: Selection Screen with Date Range of program Z_DEL_OLD_PROCESSLOG

After execution of program a message will be displayed on the screen as shown in *fig-12* about the number of records deleted from tables RSPCPROCESSLOG and RSPCLOGCHAIN.
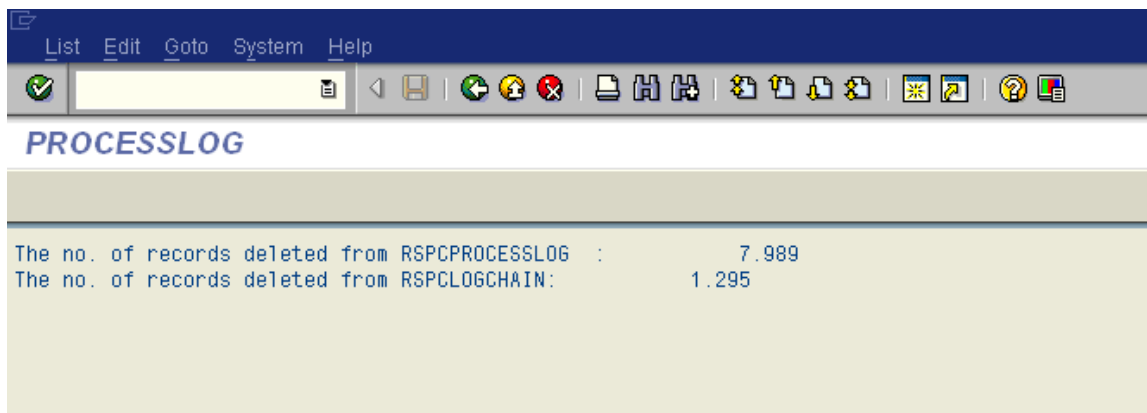
Fig-12: Output of program Z_DEL_OLD_PROCESSLOG with Date Range

This finishes execution of the program and all the unnecessary process chain execution logs are deleted.

The amount of database space saved will depend on number of records being deleted from these tables which in turn will depend upon the date selection provided by user to run this program.

The standard SAP functionality of deleting old process chain logs is not available in BW 3.1. But a functionality of deleting old process chain logs of a particular process chain is available in BI 7.0 in the form of program 'RSPC_LOG_DELETE'. But by using the above discussed program, logs of only one process

chain can be deleted in one go, but if we use above discussed program, we can delete unnecessary logs of all process chains in one go. Another problem with the program 'RSPC_LOG_DELETE' is that it can deactivate some steps in the process chain which needs to be reactivated after execution of program.

But this standard program 'RSPC_LOG_DELETE' can be integrated in our program in BI 7.0 system, so that all the standard checks done by the program 'RSPC_LOG_DELETE' would be done by our program. Please find below BI 7.0 system version of above discussed program in order to use standard program 'RSPC_LOG_DELETE'.

```
*----------------------------------------------------------------------*
*         Start of Program
*----------------------------------------------------------------------*
REPORT  Z_DEL_OLD_PROCESSLOG.
*----------------------------------------------------------------------*
*         DB-Tables
*----------------------------------------------------------------------*
TABLES: rspclogchain,    " Cross-Table Log ID / Chain ID
         rspcprocesslog. " Logs for the Chain Runs
*----------------------------------------------------------------------*
*         Variables/Internal Tables
*----------------------------------------------------------------------*
DATA: count1 TYPE p,
      count2 LIKE count1.
DATA: lv_rc LIKE sy-subrc,
      l_s_logs TYPE rspclogchain,
      l_msg TYPE char255,
      l_datum TYPE sy-datum,
      l_zeit TYPE sy-uzeit.
DATA: tb_chain LIKE rspclogchain OCCURS 0 WITH HEADER LINE.
*----------------------------------------------------------------------*
*         Selection-Screen
*----------------------------------------------------------------------*
SELECTION-SCREEN BEGIN OF BLOCK b01 WITH FRAME TITLE text-001.
SELECTION-SCREEN BEGIN OF LINE.
SELECTION-SCREEN COMMENT 1(31) text-002 FOR FIELD pa_lat.
PARAMETERS: pa_lat RADIOBUTTON GROUP rad1 USER-COMMAND click DEFAULT 'X'.
SELECTION-SCREEN END OF LINE.
SELECTION-SCREEN SKIP.
SELECTION-SCREEN BEGIN OF LINE.
SELECTION-SCREEN COMMENT 1(31) text-003 FOR FIELD pa_dat.
PARAMETERS: pa_dat RADIOBUTTON GROUP rad1.
SELECTION-SCREEN END OF LINE.
SELECTION-SCREEN BEGIN OF BLOCK b02 WITH FRAME TITLE text-004.
SELECT-OPTIONS: so_date FOR rspclogchain-datum MODIF ID abc OBLIGATORY.
SELECTION-SCREEN END OF BLOCK b02.
```

```
SELECTION-SCREEN END OF BLOCK b01.


PARAMETERS p_force TYPE rs_bool AS CHECKBOX.

*-----------------------------------------------------------------------*
*          At Selection-Screen
*-----------------------------------------------------------------------*


AT SELECTION-SCREEN OUTPUT.
  LOOP AT SCREEN.
    IF screen-group1 = 'ABC'.
      IF pa_dat = 'X'.
        screen-active = 1.
        MODIFY SCREEN.
      ELSE.
        screen-active = 0.
        MODIFY SCREEN.
      ENDIF.
    ENDIF.
  ENDLOOP.


*-----------------------------------------------------------------------*
*          Start-Of-Selection
*-----------------------------------------------------------------------*
START-OF-SELECTION.


*take complete data from RSPCLOGCHAIN into internal table tb_chain.
  SELECT * FROM rspclogchain INTO CORRESPONDING FIELDS OF TABLE tb_chain.


*sort tb_chain in descending order according to chain_id,datum,zeit
* so that latest record for particular chain id should be at top.
  SORT tb_chain DESCENDING BY chain_id datum zeit.


  CLEAR tb_chain.


  LOOP AT tb_chain.


*delete latest record from internal table so that it should contain only
*those records which we need to delete from both tables.
    ON CHANGE OF tb_chain-chain_id.
* keep latest records only
      IF pa_lat = 'X'.
```

```
        DELETE tb_chain.
* delete only those records which meet date range selection option
      ELSEIF pa_dat = 'X'.
        DELETE tb_chain WHERE chain_id = tb_chain-chain_id
                        AND   datum NOT IN so_date.
      ENDIF.
    ENDON.


  ENDLOOP.


  LOOP AT tb_chain INTO l_s_logs.


    CALL FUNCTION 'RSPC_LOG_DELETE'
      EXPORTING
        i_logid      = l_s_logs-log_id
        i_force      = p_force
      EXCEPTIONS
        failed       = 1
        no_authority = 2
        OTHERS       = 3.
    IF sy-subrc = 2.
      MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
              WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4
              INTO l_msg.
      WRITE / l_msg.
    ELSEIF sy-subrc <> 0.
      IF p_force = 'X'.
        MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
                WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4
                INTO l_msg.
        WRITE / l_msg.
      ELSE.
        ROLLBACK WORK.
        MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
                WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
      ENDIF.
    ELSE.
      CALL FUNCTION 'RSSM_GET_TIME'
        EXPORTING
          i_datum_utc = l_s_logs-datum
          i_uzeit_utc = l_s_logs-zeit
```

```
        IMPORTING
          e_datum_loc = l_datum
          e_uzeit_loc = l_zeit
        EXCEPTIONS
          failed      = 1
          OTHERS      = 2.
      IF sy-subrc <> 0.
        l_datum = l_s_logs-datum.
        l_zeit  = l_s_logs-zeit.
      ENDIF.
      MESSAGE s129(rspc) WITH l_s_logs-log_id l_datum l_zeit INTO l_msg.
      WRITE / l_msg.
    ENDIF.


  ENDLOOP.


*-------------------------------------------------------------------------*
*         End of Program
*-------------------------------------------------------------------------*
```

Generally this program should be used in 3 months to delete 3 months before old logs of process chains, as these logs won't be required normally.

Related Content

https://www.sdn.sap.com/irj/sdn/weblogs
http://help.sap.com/

http://www.tli-usa.com/download/TipsTricks_and_Techniques_for_OptimalUseofProcessChains.pdf

For more information, visit the Business Intelligence 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.