Retrieving Archived Data to Report Program.

Applies to:
Developing Archiving Solutions, for more information visit Archiving Solutions

Summary
This document provides details on how to retrieve archived data from archive files to report program. We will also see why and what data needs to be archived and what is the role of information structure (INFO STRUCTURE) in retrieving archived data.

Author: Manoharan Murugesan
Company: Intelligroup Asia Pvt Limited
Created on: 29 April 2009

Author Bio
Manoharan Murugesan works with Intelligroup Asia Pvt Ltd since 2008. He has good knowledge in SAP ABAP.
Table of Contents

Introduction .........................................................................................................................................................3
Benefits ...............................................................................................................................................................3
Which data to be archived? ................................................................................................................................4
How data will be archived? ................................................................................................................................4
  Info Structure...................................................................................................................................................4
Assumption .........................................................................................................................................................5
  Step by step procedure for retrieving archived data .......................................................................................5
Related Content................................................................................................................................................14
Disclaimer and Liability Notice..........................................................................................................................15
Introduction

As the amount of data stored in SAP increases rapidly with day to day transactions in production environment, there has to be a way to control the growing data. Data archiving is the solution provided by SAP to keep the size of database under control. Data archiving also ensures long time access to your data. Archive Development Kit (ADK) is a service provided by SAP and it acts as a basis for data archiving solutions. ADK is an API between SAP applications, database and archive files.

Benefits

- By reducing the size of the SAP database, system performance can be improved. Data growth in the SAP system adversely affects performance.
- Certain data cannot be just deleted because of taxation (IRS) regulations, FDA requirements, government regulations, internal organizational requirements, and audit requirements.
- To reduce the amount of time and effort required for systems administration back-ups, client copy, upgrades, month-end processes, etc.

Above mentioned benefits can be illustrated with the help of an example, consider an SAP system is like a train with several carriages being pulled by the engine.

If the carriages that exist are full, the reaction is to add additional carriages. This process is only being a short term solution as the engine just has enough power to pull a prescribed load after which it begins to stall. Getting some of the load off the train is a more logical solution; and the earlier this process is undertaken the greater the benefits. There will be savings in the purchase of new carriages or even having to add a new more powerful engine.
Which data to be archived?

Data which are accessed less frequently and no longer required for daily operations will be archived. Usually transaction data will be archived periodically on a scheduled basis.

How data will be archived?

First thing one should know while proceeding for archiving is the central component of data archiving i.e Archiving Object. Database tables cannot be archived directly because they don’t know the business context of the data that needs to be archived. We can create significant Archiving objects like accounting, sales, controlling, purchasing etc and assign related tables to respective objects so that all sales or accounting or controlling related tables will be archived together.

Info Structure

In order to read data from archived files a structure is required. It is this structure through which data from archived files reaches the report program; this structure is known as information structure (INFO STRUCTURE). INFO STRUCTURES are created for a specific Object name and field catalog combination.

Fields in field catalog are filled based on the tables that get associated with an archiving object. Then the field catalog is assigned to that particular Archiving object in transaction SARJ menu->Environment->Field Catalog.
Note: archiving data to archive files has different steps to follow. We are dealing with retrieving data from archiving files to report program.

**Assumption**

Consider a database table COEP was archived; with data from the year 1997 to 1998.

**Step by step procedure for retrieving archived data**

- First step, identify the object name for the archived table. Important transaction used to find the object name is DB15.
- As shown in the below screen shot, table name can be entered and all Archiving objects associated with that table can be viewed and vice versa.

  ![Screen Shot](image)

- Object name and the corresponding INFO STRUCTURE name can be seen in transaction SARE (Archive Explorer).
In this case, the relevant Object name and the corresponding INFO STRUCTURE name for table COEP is shown in the below screen shots.

Note: Relationship between Archiving object and INFO STRUCTURE are established while creating an INFO STRUCTURE in SARJ T-code.

- Once the object name and INFO STRUCTURE name is found for the archived table, a read program will be used to read both archiving object and archiving class data from archive files.
- INFO STRUCTURE has number of fields and key fields for an INFO STRUCTURE will be set at the time of configuration. The INFO STRUCTURE fields can be seen in SARJ transaction.
- Field names with lock symbol on the left (Shown in the below screen shot) are key fields of an INFO STRUCTURE. And these fields can be used as selection condition to select archiving objects.
In order to select data from archived files based on selection inputs (Selection Screen input field values of report program), function module ‘AS_API_READ’ is used. Exporting parameters i_fieldcat and i_selection is used to pass ‘INFO STRUCTURE NAME’ and selection inputs respectively.

Exporting parameter i_selection gets all selection screen input values through an internal table gt_selections[]. The below screen shots shows how gt_selections[] is populated with selection screen input values.
Note: In some cases, certain selection screen values cannot be passed directly to gt_selections[]. In our example, select-options posting year (POST-YR) and posting period (POST-PER) are directly used for populating gt_selection[]. But select-options cost center and cost element are not used directly, they are converted into lr_objnr (Object number) and used as shown above. For converting selection screen values to lr_objnr (Object number), see the standard include: RKAEPFOB (IN FORM Get_cost_centers).
Based on the above inputs, the importing parameter e_result populates an internal table.

The internal table will have archive key and archive offset values as shown below.
Now based on these archive key and archive offset values, we can read all associated archive data objects from all opened archived files.

To read all object from opened archived files, call the function module 'ARCHIVE_READ_OBJECT' and pass values to the exporting parameters Object, Archive key and offset.

The importing parameter Archive_handle will set a value for performing read operation.
Retrieving Archived Data to Report Program.

Note: The above function module 'ARCHIVE_READ_OBJECT' should be called inside a loop statement to read each object opened in the archived files based on the archive key and archive offset values.

- Next step is to read each record from an object and to do this, call function module 'ARCHIVE_GET_NEXT_RECORD' in a loop within the loop started earlier.
- Pass the archive handle value to the exporting parameter Archive_handle.
- The importing parameter 'Record' will give the values of the selected record and parameter 'Record_structure' will give the structure of the selected record.
• Declare an internal table with the required structure and populate the internal table whenever 
Record_structure is equal to the structure of the declared internal table.
• End the loop opened for reading each record of an object.

Note: CASE statement can be used to populate the internal table. For each loop pass, the case statement 
checks for ‘l_structure’ value and the corresponding internal table gets appended with values in ‘lit_data’ as 
shown below.

```sql
CALL FUNCTION 'ARCHIVE_GET_NEXT_RECORD'
  EXPORTING
    archive_handle = g_read_handle
  IMPORTING
    record = lit_data
    record_structure = l_structure

EXCEPTIONS
case 'COEP'.
  ASSIGN it_coep TO <fs_struc> CASTING.
  <fs_struc> = lit_data.
  APPEND it_coep.
endcase.
ENDCASE.
```

```sql
REFRESH lit_data.
CLEAR g_object_cnt.
ENDDO.
```
• Call function module 'ARCHIVE_CLOSE_FILE' and pass the archive handle value to exporting parameter Archive_handle. This function module will close the archive session for an object.

```plaintext
CALL FUNCTION 'ARCHIVE_CLOSE_FILE'
  EXPORTING
      archive_handle = g_read_handle
  EXCEPTIONS
      internal_error = 1
      wrong_access_to_archive = 2
      OTHERS            = 3.
    IF sy-subrc <> 0.
      MESSAGE i004. "Unable to close archive session".
      EXIT.
    ENDIF.
  ENDOFF.
```

• Step 10 to 17 will be repeated for each object opened in the archived files (based on archive key and offset value).

• End the loop opened for reading each object opened in the archived files.
Related Content

Procedure For Developing Read Program
Archive Development Kit
Archive Retrieval Configurator

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.