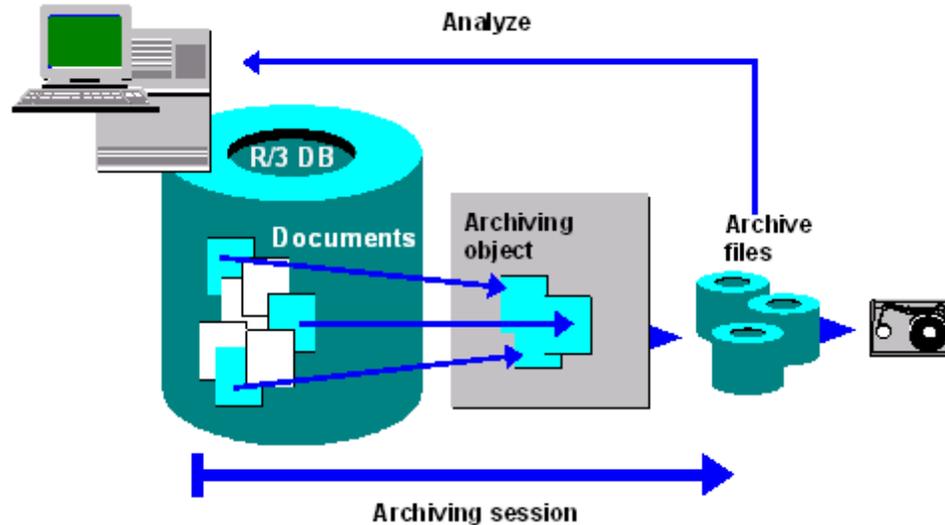


Introduction to Data Archiving (CA-ARC)

Data Archiving removes mass data from the database that the R/3 System no longer needs online, but which must still be accessible at a later date if required. The following graphic illustrates the archiving process: Archiving objects are used to write documents to archive files, which can be stored on other media.



Data in the R/3 Database can only be archived using **archiving objects**, which describe the data structure and context.



Financial Accounting documents are, for example, archived using the archiving object FI_DOCUMNT. It includes the document header, company code-dependent postings, change documents, SAPscript texts, and other elements.

Integration

The SAP Data Archiving concept is based on the Archive Development Kit (ADK). The ADK provides the technical basis for the archiving transaction (SARA). To call the archiving transaction, choose *Tools* → *Administration* → *Management* → *Data Archiving*, or directly from the application component. If archive management is called from the application component, application-specific settings (such as programs and archiving objects) are activated automatically.

Archiving objects for each application component are predefined in the system. Their structures are described in the application-specific sections.

Features

The archiving procedure is divided into three main steps:

- **Create archive files:** The data to be archived is written sequentially to a newly created archive file.
- **Store archive files:** The newly created archive files can then be moved to a storage system or copied to a tape. The removal to an external storage system can be triggered manually or automatically.
- **Run delete program:** The delete program reads the data from the archive files and then deletes it from the database.

You can schedule archiving programs as background tasks or run them during normal online operations

Reasons for Archiving

There are both technical and legal reasons for archiving application data. SAP Data Archiving:

- Resolves memory space and performance problems caused by large volumes of transaction data
- Makes master data easier to handle and to keep up-to-date
- Ensures statutory data retention rules are observed
- Ensures that data can be reused at a later date, for example, in new product development

Archiving Requirements

Data archiving is intended to more than just save the contents of database tables. Archiving must also meet the following requirements:

- **Statutory requirements**
Data must be archived in such a way that it can be analyzed at any time in the future. In some countries, such as the USA, the tax authorities require that the data be stored transparently so they can analyze the data with their own computer systems.
- **Hardware independence**
Because the encoding of numerical fields, such as integers depends on the hardware being used, archiving must ensure that information about hardware is appended to the archived data so that the data can be displayed later with different hardware.
- **Release dependence**
Because the data structure may depend on the version of R/3 you are using, record structure and field definition information must also be archived.
- **Data dependencies**
Data objects often depend on other data objects. Consequently, in order to remove a specific data object from the system, you must determine whether other data objects must already be archived, or must be archived at the same time.
- **Enterprise and business structure**
Some data is only of use if you are familiar with an enterprise's organizational structure, such as

how it is divided into different areas for sales. Archiving must therefore ensure that this information is also archived.

The above list shows that data archiving requires a detailed knowledge of data semantics. For this reason, R/3's application-integrated archiving approach is superior to archiving products that are not integrated into R/3 (database tools).

Basic Archiving Terms

This section defines some basic SAP Data Archiving terms:

Optical Archiving

Reorganization

Residence and Retention Periods

Backup/Restore

Optical Archiving

When using a storage system, it is important to differentiate between the storage of original documents, such as scanned incoming invoices, and the storage of archive files that contain application data.

Document archiving means storing scanned original documents, outgoing documents, or R/3 printlists in an attached storage system. You can display these documents and lists at a later date, but you cannot analyze them with R/3 tools or reload them into the R/3 System.

Archive files containing data objects that were created by SAP Data Archiving can, with the help of the Content Management Service (CMS), be transferred to an external storage system and be made available in the file system.

Reorganization

The term **reorganization** comes from R/2, where data was removed and then reloaded in order to optimize the physical data distribution on the disk. During this procedure, unnecessary data was often archived and deleted. Reorganization has nothing to do with SAP Data Archiving using the Archive Development Kit (ADK)

Residence and Retention Periods

The **retention period** is the entire time that data spends in the database before it is archived. The **residence time** is the minimum length of time that data **must** spend in the database before it meets the archivability criteria.



If the residence time is a month, data that has been in the system for two months will be archived. Data that is only three weeks old remains in the database.

Backup/Restore

A **backup** is a copy of the database contents that is made so you can restore the database if part or all of the data is lost or damaged during a system breakdown. Backups are usually made at regular intervals according to a standard procedure. Reloading the saved data into the file system is called **restoring** the data.



Releases 2.1 and 2.2 of R/3 required you to backup the database before archiving data. This safety measure is no longer required from Release 3.0.

Archiving Features

The key features of SAP Data Archiving are:

Data Security During Archiving

Link to Optical Archive

Data Compression

Archiving Without Database Backup

Accessing Archived Data

Converting Old Archive Files

Data Security During Archiving

Archiving is a two-step procedure (a third step, the archive file storage, is optional). In the first step, the data is written to archive files. In the second step, the system reads the archived data from the archive files and deletes it from the database.

This two-step process guarantees data security if problems occur during the archiving process. For example, the procedure identifies network data transfer errors between the database and the archive file. If an error occurs, you can restart the archiving process at any time because the data is still either in the database or in an archive file. This means that you can usually archive *parallel to the online application*, that is, during normal system operation, without having to back up the database first.

You can further increase data security if you store the archive files in an external storage system before you delete the data from the database. This guarantees that the data from the database will only be deleted after it has been securely stored in an external storage system. You can define the storage time in archiving object-specific Customizing.

Data Compression

During archiving, data is automatically compressed by up to a factor of **5**. However, if the data to be archived is stored in cluster tables, no additional compression takes place.

Memory Space Gained by Archiving

Increased storage space in the database and the resulting performance gains in the application programs are the most important benefits of data archiving. Therefore it is useful to know how much space the data to be archived takes up in the database.



Note that *cluster* tables are stored in a compressed form in the database.

It may also help to know in advance how much space the archive files that you create will need.



Data is compressed before it is written to the archive file. The extent of the compression depends on how much text (character fields) the object contains. Optimal compression can achieve a **factor of five**.

Estimates of the size of an individual data object are provided in the application-specific sections of this documentation. You can, however, also make a **rough approximation** of how much space the data to be archived occupies in the database and how much space the archive files will require.

Archiving Without Database Backup

You can execute SAP Data Archiving without backing up the database first. However, we recommend that you backup archive files before storing them

Accessing Archived Data

Because archived data has only been removed from the database and not from the application component itself, the data is always available. Archive management allows three types of access:

1. (Read) access to a single data object, such as an accounting document
2. Analysis of an archive file (sequential read)
3. Reload into the database (not possible for all archiving objects)

Converting Old Archive Files

When archived data is read, the system automatically makes the conversions required by hardware and software changes.

When old archive files are accessed, the Archive Development Kit (ADK) can make allowances for changes to database structures (field types, field lengths, new fields, and deleted fields) after the data was archived and for changes to hardware-dependent storage formats. This is only done during read access. The data in the archive file is not changed. The following items are changed (if necessary) during automatic conversion:

- Database table schema (new and deleted columns)

- Data type of a column
- Column length
- Type of file (ASCII, EBCDIC, and so on)
- Number format (such as the use of the integer format on various hardware platforms)

If database structures in an application have undergone more changes than the ADK can handle (for example, if fields have been moved from one table to another or if one table has been divided into several tables), then a program is provided by the application for the permanent conversion of existing archive files

Link to External Storage System

Content Management Service

Archive files created by SAP Data Archiving can be stored on tertiary storage media, such as WORMs, magnetic-optical disks, and tapes using the Content Management Service (CMS). This can be done manually or automatically.



From Release 4.6C, the Content Management Service replaces the SAP ArchiveLink in the Data Archiving context.

Hierarchical Storage Management Systems (HSM Systems)

Archive files can also be transferred to an HSM system simply by storing them in the file system of the HSM system. For storage, the HSM system can also use tertiary storage media, such as MO-disks.

The Archiving Procedure

Purpose

This process describes the basic archiving procedures.

Process Flow

Archiving is carried out in three steps:

Step 1: Create Archive Files

Step 2: Store Archive Files

Step 3: Execute Delete Program

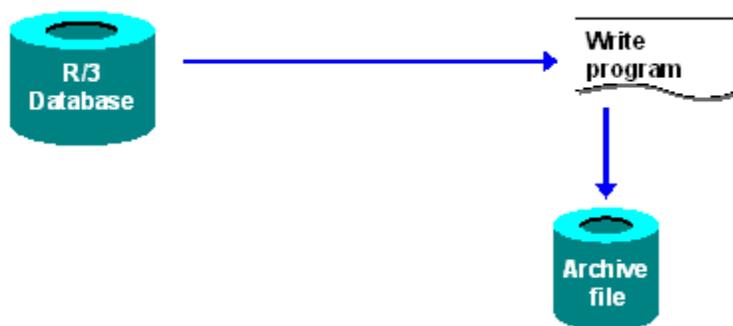


You can also opt to store the archive files after the deletion step: To do this you must flag the checkbox *Deletion run before file storage* in archiving object-specific customizing.

If security is your main concern, then you should not run the deletion program until after the archive files have been stored. In this way you know that the data will only be deleted from the database after the archive files have successfully been moved to the external storage system. In addition, you can set the system to read the data from the storage system and not from the file system.

However, if your main concern is the performance of the archiving programs, then you should schedule the deletion program first and then store the files.

Step 1: Creating (the First) Archive Files



In step one, the archiving program creates an archive file. The data to be archived is then read from the database and is written to the archive file in the background. This process continues until one of following three events occurs:

1. All the data is written to an archive file
2. The archive file reaches the maximum size specified in Customizing
3. The archiving is not yet finished, but the archive file contains the maximum number of data objects specified in Customizing



If in cases 2 and 3 there is still data to be archived, the system will create another archive file.

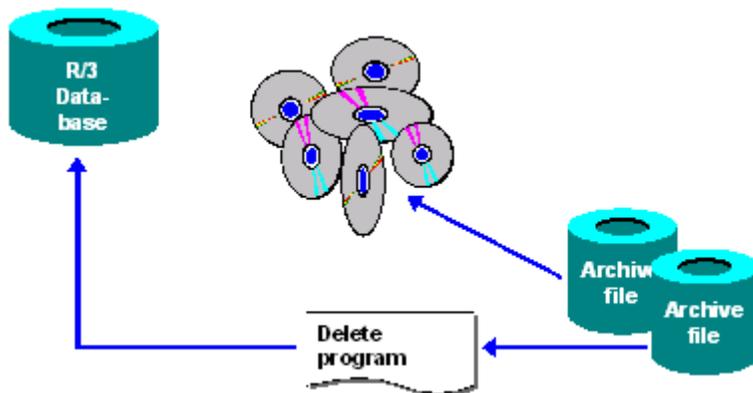
If the event 2 or 3 occurs, then archive management continues with the delete step (assuming that the following checkboxes are flagged in archiving object-specific customizing: *Start delete program automatically* and *Delete before storage*).

Write Run Followed by Event

After all archive files have been completely created for an archiving session, the ADK triggers the system event SAP_ARCHIVING_WRITE_FINISHED. By reacting to this system event, you have the option of scheduling archiving jobs automatically. This includes the rebuilding of indexes or input help, archive file backup before deletion, and so on. The event parameter is the number of the archive session.

You can define new jobs in transaction *SM36*. You maintain events in transaction *SM62*.

Step 2: Storing Archive Files



Once created by the write program, archive files can be stored in various ways.

- **Content Management Service**

If a storage system is connected to the SAP system, and the write run has completed

successfully (the corresponding settings are contained in [Archiving Object-Specific Customizing](#)) the storage system is instructed to store the archive files. You can also store archive files manually.

- **HSM systems**

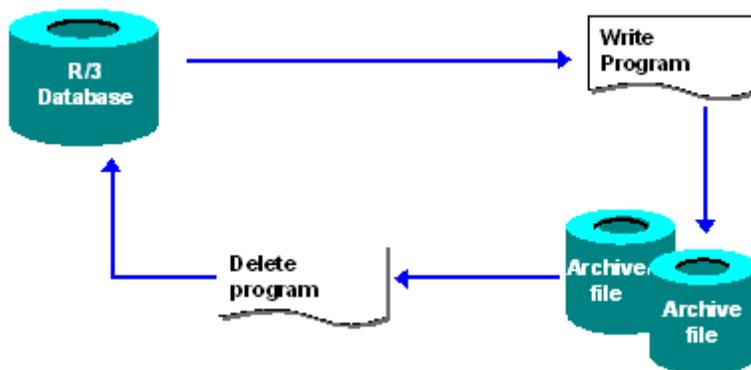
If you maintain the file path of an HSM system in Customizing (transaction *FILE*), you don't need to communicate with the archive system using ArchiveLink because the HSM system stores the files autonomously.

- If you use an HSM system, it is sufficient to maintain the file name in Customizing (Transaction *FILE*). You do not then need to communicate with the storage system using the Content Management Service, because the HSM system stores the files on tertiary media according to access frequency and disk space.

- **Manual storage**

Once the delete program has processed the relevant archive file, you can manually copy archive files to tape.

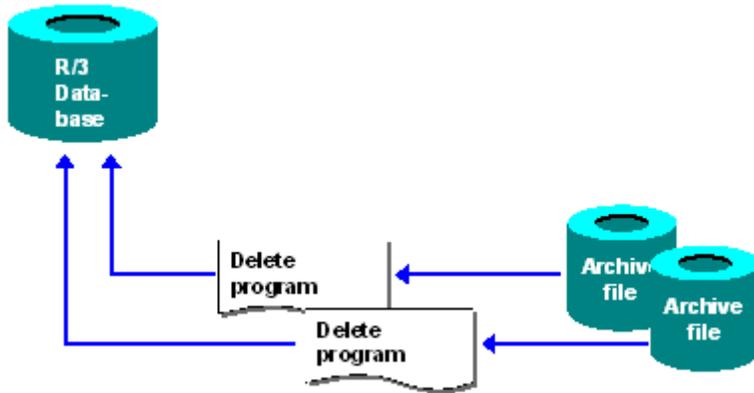
Step 3: Running Delete Programs



After closing the first archive file, the archive management system creates a new archive file and continues with the archiving process. While this happens, another program reads the archived data from the completed archive file and deletes it from the database. This procedure guarantees that only data that has been correctly saved in the archive file is deleted from the database.



If you do not carry out deletion until after the data has been stored, you can make a setting in Archiving Object-Specific Customizing so that the system will read archive files from the storage system during deletion. In this way, you can detect errors in good time which might arise when transferring or saving the archive files in the storage system.



When the last archive file is closed, a delete program is also executed for this file. The graphic shows that several delete programs are running simultaneously. Because the archiving program makes no changes in the database, it creates new archive files faster than the delete programs can process them. This decreases the total archiving runtime because the database is used more efficiently.

Delete Run Followed by Event

After all archive files have been completely created for an archiving session, the ADK triggers the system event `SAP_ARCHIVING_DELETE_FINISHED`. By reacting to this system event, you have the option of scheduling archiving jobs automatically. This includes the rebuilding of indexes or input help, archive file backup before deletion, and so on. The event parameter is the number of the archive session.

You can define new jobs in transaction `SM36`. You maintain events in transaction `SM62`.



You can also have *event-controlled* delete runs started automatically. You can define the event that triggers the delete program in the group box *Settings for the Delete Program* in Archiving Object Specific Customizing.

The Archiving Object

Definition

Archiving objects are the core of the SAP Data Archiving concept. The archiving object specifies which data is archived and how. It also describes which database objects must be handled together as a single business object and which must be interpreted independently.

Use

An archiving object has a name of up to ten characters in length. You need this name to access archiving for the object in archiving management. However, the function for calling archiving management is often integrated in the application menu, where the archiving object name is set as the default so that you do not need to enter it yourself.

Structure

The following programs must (or can) be assigned to an archiving object. (Archiving objects are defined using transaction *AOBJ*)

The SAP System contains programs (some of which are optional) for the following actions:

1. **Preprocessing** (optional)

Some archiving objects require a program that prepares the data for archiving. This program marks data to be archived, but it does not delete any data from the database.

Preprocessing programs must always be scheduled manually and are run from archive administration.

2. **Write**

This program creates archive files and writes data to them. It does not delete the data from the database.

You can specify in archiving object-specific Customizing whether the next phase (deletion) is to take place automatically after the archive files have been created. Delete jobs can also be event-triggered. To do this, you set up the trigger event in archiving object-specific Customizing.

3. **Delete**

This function can include several processes. The processes are always dependent on the existing archive files.

The data is usually deleted from the database, but in some circumstances, the archived data is only marked for deletion in the database.

In archiving object-specific Customizing, you can specify that archive files, after successful processing, are to be transferred to an external storage system using the Content Management Service.

- **Postprocessing** (optional)

The *Postprocessing* function can be performed asynchronously with the *Delete* program. This function is not available for all archiving objects. If the data has not yet been deleted from the database by the delete program, it is deleted by the postprocessing program. The postprocessing program can also perform other tasks. For more information, see the documentation for the specific object.

- **Reload archive** (optional)

You can reload archived data from the archive files into the database using this function. It is not available for all archiving objects. To access this function, choose *Goto* → *Reload*.

- **Index** (optional)

This function constructs (or deconstructs) an index that allows individual access. It is not included in every archiving object

Sample Data Description Used by the FI Document

The archiving object data structure is described here using Financial Accounting as an example. To archive FI documents so that they can be accessed in the future, you must make sure that the environment is stored in the archive file. This is ensured by the respective data description.

The archiving object for FI documents (*FI_DOCUMNT*) includes the following data:

- Document header
- Document segment accounting
- Document segment control data
- Posting procedures that depend on company codes
- Document segment variable data
- Document segment change data
- Change documents
- SAPscript texts

The Data Object

Definition

A **data object** is the application-specific instance of an archiving object, that is, an archiving object filled with concrete application data. The archiving object only describes the data structure, that is, it determines which data belong logically together, whereas the data object contains real data from the database, such as all table entries belonging to an FI document.

Use

The ADK ensures that data objects are written sequentially to an archive file. All data objects in an archive file have the same structure, which is described in the archiving object.

Archive Administration

Use

All interaction relating to data archiving takes place in the Archive Administration transaction (SARA)

The definition of an archiving object determines which functions are possible. Consequently, all archiving functions are not always available in the Archive Administration.

Features

Archiving

Enables a write program to be scheduled and run, whereby data is removed from the application database. The program writes the relevant data sequentially in archive files. See [Creating Archive Files](#)

Deletion

Enables a deletion program to be scheduled and run, whereby data objects, which could be read from the previously created archive file, to be removed from the database. See [Deleting Data from the Database](#)

Evaluation

Enables a program to be scheduled and run that analyzes archived data. See [Analyze Archive Files](#)

Index

Enables an index to be constructed or deconstructed for existing archive files. The index is required to display individual documents belonging to several archiving objects. See [Constructing an Index](#) and [Deconstructing an Index](#)

Storage System

Enables archive files to be transferred to a connected storage system and enables stored archive files to be retrieved from the storage system. See [Storing Archive Files](#) and [Retrieving Stored Archive Files](#).

Management

Offers an overview of archiving sessions. From here, you can display and analyze object-specific management information. See [Archive Administration: Overview of Archiving Sessions](#) and [Calling the Archive Session Management](#)

Preprocessing

Enables you to schedule and run preprocessing programs. Preprocessing programs prepare data objects for archiving, for example, by setting a deletion indicator. See [Scheduling Preprocessing](#).

Postprocessing

Enables you to schedule and run postprocessing programs. Postprocessing programs carry out operations following an archiving session, such as updating statistics. See [Scheduling Postprocessing](#).

Further Functions

Depending on the action you have selected, you can access the following menu options by choosing *Goto*.

- [Network Graphic](#)
- [Reloading](#)
- [Customizing](#)
- Job overview: Provides an overview of all archiving jobs and the functions available to process them.
- [Management](#)
- Stored files: Enables you to search for stored archive files according to selection criteria.
- [Database Tables](#)
- [Infosystem](#)

Activities

You can call the Archive Administration:

- By entering the transaction *SARA* in the command field
- By choosing *Tools* → *Administration* → *Management* → *Data Archiving*
- Directly from the application

If you choose an archiving function in the application, the relevant archiving object is automatically specified in Archive Administration. Otherwise, you must specify it manually

The Archive Development Kit (ADK)

Definition

Archive management is based on the Archive Development Kit (ADK), which is located between the application and the archive. The ADK provides all the functions required for archiving data and accessing archived data.

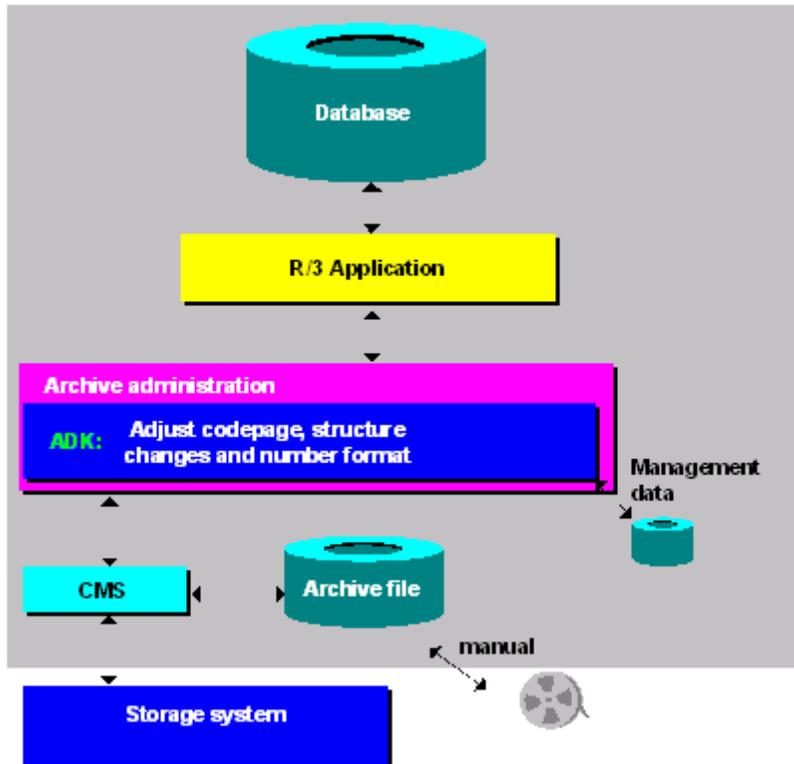
Use

The ADK functions are required for archiving and subsequent access to archived data. The ADK automatically performs the hardware-dependent adjustments (such as codepage and number format) and structural changes that are required when archive files are created. It also temporarily converts data that was archived with previous R/3 releases (from 2.1) when that data is accessed later.

Integration

The following graphic illustrates the ADK and archive administration concept.

www.agconsultraining.com



User Authorization Checks

Definition

An authorization object (*S_ARCHIVE*) controls access to the programs for an archiving object. The ADK performs a check when an archive file is opened for one of the following actions:

- Write
- Delete
- Read
- Reload

Use

The following authorizations can be given per archiving object and application component, (such as FI, BC):

- **Everything is allowed**
Write, read, and reload archives; execute delete programs; change mode in archive management

- **Change mode in archive management**
Maintain notes
- **Read and analyze archives and display mode in archive management**

There may also be additional access authorization checks for specific application components

DB Tables

Use

This function (transaction *DB15*) enables you to define all the archiving objects that archive the contents of a specific table, or enables you to display all of the tables for a specific archiving object. Furthermore, you can display information on storage space in the database or report existing statistics.

Scope of Functions

Archiving objects

You can determine all archiving objects belonging to a specific table.

Tables from which you are going to archive data

You can determine all tables that are linked to a specific archiving object, and from which data records will be deleted during an archiving session.

Storage space (online)

This transaction enables you to obtain detailed information on occupied storage space and additional general information on archiving sessions.

Storage space (statistics)

This transaction provides information from SAP tables that are filled by statistics determination runs.

DB15 is a *Computer Center Management Systems (CCMS)* transaction

Maintain Start Date and Spool Parameters

Use

You can maintain the start date and the spool parameters on the following screens:

- Create archive files
- Preparation
- Postprocessing
- Reload archive
- Start analysis program

Features

The *Start date* and *Spool parameters* pushbuttons are followed by a traffic light icon and a note. The table lists the three possibilities.

Traffic Light	Note text	Meaning
	Not maintained	The values are not maintained
	Maintained	The values have already been maintained in this session, for the current action.
	Maintained	The values are maintained.

The Network Graphic

Use

The network graphic displays the dependencies between archiving objects and schedules archiving sessions.

Features

You must account for dependencies between archiving objects in an archiving session. In general, you cannot archive data for an archiving object that has preceding objects until these preceding objects have been archived.

The network graphic tells you whether the archiving object has preceding objects that must be archived first. The nodes in the network graphic represent the archiving objects. A node displays the following information:

- Archiving object name
- Application component name
- Short description
- Date of the last archiving
The date is displayed in one of three colors:

- Green
Archiving and deletion successful
- Yellow
Successfully archived, but not yet deleted, or
Archiving still running, or
Deletion running, or
Deletion cancelled
- Red
Not yet archived, or
Archiving cancelled

Activities

You can use the network graphic to call archive management at the same time as the required object name:

1. To call archive management, double-click on the action you want to perform (for example, archive or delete). You access archive management. The name of the archiving object is copied automatically.
2. You can now choose the action that you want to carry out and create the relevant background job

The Standard Log

Definition

A **standard log** is usually created during the archiving session, but you can also create a log for application components. See the documentation on specific archiving objects for more information.

Structure

The standard log contains the following information on each archiving session or archive file:

- Number of data objects archived
- Tables involved
- Number of table entries processed
- File size

You can call the standard log from the screen *Archive Administration: Overview of Archiving Sessions*. Choose *Spool List*

Accessing Archived Data

Use

Data that has been archived using SAP Data Archiving has only been removed from the database, and not from the application component. Data is still available for read access and analysis. In some cases, archived data can even be reloaded into the database.

Prerequisites

A prerequisite of read access and reload access, is that the file can be found in the file system. You can check this in archive administration.

Features



The uses of archived data described here are technically possible, but are not currently implemented in all application components. For more information, refer to the specific application components.

Three types of access are possible:

- **Read access to a single data object (such as a posting document)**
Direct access requires an index that can be constructed either during or after archiving . A complex search of the documents stored in the archive files, in which all orders of an article in a particular batch are required for a product recall action, is not possible.
- **Analysis of the archive file (sequential read)**
The analysis can be of one or more archiving sessions. You can also link current data in the database and archived data.
- **Reload to the database**
Archived data does not usually need to be reloaded because it remains accessible by the applications. There is also a lot of data that cannot be reloaded or for which reloading is inappropriate. For this reason, reload programs do not exist for all archiving objects

Customizing

Purpose

Customizing for Archiving enables you to set parameters that affect the archiving process.

Prerequisites

- You have customizing authorization.
- Data types are maintained.
- Organizational data is maintained.

Process Flow

Customizing can be divided into the following areas:

- [General Customizing](#)
- [Cross Archiving-Object Customizing](#)
- [Archiving Object-Specific Customizing](#)
-



The application chapters contain information on customizing specific archiving objects.

General Customizing

Purpose

General Customizing is usually performed when the SAP System is installed.

Process Flow

All settings can be made with the Customizing function. You have the following options:

1. You make Customizing settings (such as setting the residence time) for specific application components.



Criteria for archiving and deleting data for specific archiving objects are specified in Customizing for that object. Note that such criteria may not be set for all objects. The application component-specific sections of this documentation contain information on how to set the criteria.

2. You perform Customizing for Basis

Cross-client and client-specific platform-independent file names are maintained Customizing for Basis (see [Defining Logical File Names](#)).



You are advised to maintain the platform-independent file names as cross-client.

If you want to transfer files to a storage system using the Content Management Service (CMS), ensure that the correct Content Repository is set up in archiving object-specific Customizing

Cross-Archiving Object Customizing

Use

The settings that you make here apply on a cross-application and cross-archiving object basis for all archiving objects.

Features

Data archiving monitor

Use this checkbox to activate or deactivate the data archiving monitor (transaction *SAR_SHOW_MONITOR*)

.

If you mark this checkbox before data archiving, archiving-relevant information on the write and deletion runs is updated. This information can be analyzed using the data archiving monitor. If there are errors, alerts are issued.

The data archiving monitor offers the following information:

- Overview of all the archiving objects that have been run
- Detailed information on the individual archiving sessions
- Processing status display
- Help on analyzing open alerts

The data archiving monitor is a part of SAP CCMS-monitor collection (transaction *RZ20*) and can be found under *SAP CCMS Monitor Templates → Database → Data Archiving*.

For more information, refer to  [Data archiving monitor](#).

Access check for archive selection

When selecting files for reading, deleting, analyzing or reloading, this function enables you to check for the existence of an archive file, that is, whether this archive file is accessible in the archive administration. You can use this function to check for stored archive files and archive files that still exist in the file system.

- By marking the *For Files in File System* checkbox, you specify the access check covers all archive files in the file system.
- By marking the *For Stored Files* checkbox, you specify that the access check covers all archive files in a storage system.

If an archive file is not found by the access check, the file is marked as "not accessible" in the selection screen.



In the case of an access check for stored files, access to the storage system is required. The check can therefore be very time consuming and should only be used if strictly necessary.

Verification of archive files

You can use this function to specify whether, during the writing phase, archive files are to be given verification information that is to be analyzed at a later point. Storing verification information does not affect the size of the archive files. The availability of verification information is a prerequisite for the subsequent evaluation when deleting, reading or reloading.

The system uses this information to verify the status of the archive files before accessing them. Incorrect files are recognized and reported immediately, and deletion from the database does not start.

You can specify whether the verification information is to be stored, and at what point (during deletion, read or reload phase) the information is to be analyzed.

- By marking the *Create Verifiable Files* checkbox, you specify that verification information is stored during the writing phase.
- By marking the *Deletion, Read or Reload* checkboxes, you specify the point at which the verification information is to be analyzed. This can be done during the deletion, read or reload phases

Archiving Object-Specific Customizing

Purpose

You can set parameters that apply only to a specific archiving object.

Process Flow

All parameters can be set using the *Customizing* function. You can set the following parameters:

- [Logical File Name](#)
- [Archive File Size](#)
- [Deletion Program Settings](#)

- [File System to Storage System](#)
- [Postprocessing Settings](#)
- [Server Selection](#)

Logical File Name

Use

Here, you specify which logical file name the archiving object is to use when storing the archive files in the file system. Both the physical name and the path - under which the archive file is created in the file system - are derived from this logical file name.

Features

Provided you have not defined a logical file name that differs from the standard, you do not need to make any further settings. For more information, refer to [Defining logical path and file names](#).

Archive File Size

Use

This parameter specifies the maximum size that an archive file can reach when running a write program.

Features

You can specify:

- Maximum size in MB or
- Maximum number of Data Objects

The value that is reached first triggers the creation of a new archive file. If you leave both fields blank, a new archive file is created.

Note that the maximum size of an archiving file is limited by the operating system and also by the storage system if one is connected using the *Content Management Service*.



Since the deletion program is automatically started for each archive file, you can carry out deletion and archiving activities in parallel if relatively small archive files are created. This parallel activity can have a positive effect on the runtime of an archiving session, as the database is used more efficiently. If a file is too small, the number of processes rises and the system is negatively affected.



The ideal size is between 20 and 100 MB. The size must not exceed 2 GB, or no index can be constructed, thereby preventing single document access

File Storage to Storage System

Use

Here, you can specify whether an archive file is to be transferred automatically after successful processing to a connected storage system. The storage is controlled by the *Content Management Service* (CMS). You can also specify the sequence of the deletion and storage phases.

Prerequisites

A storage system is connected to the SAP System.

Features

To start the automatic storage of archive files, mark the checkbox *Start Automatically* and choose the required Content Repository.

Under Sequence, you can specify when the storage is to take place:

- If you mark *Delete Before Storage*, the archive file is not stored until it was processed by the deletion program. If, however, the deletion program is run in test mode, automatic storage is not carried out. This option is advantageous from a performance point of view.

With this setting, you can also perform storage manually if no deletion program has been run. In this case, the system informs you that this does not match the selected sequence. You must then ensure that the file is not stored at the same time as the delete program is active.

- If you mark *Store Before Delete*, the archive file is stored as soon as the write program has created the archive file, but still before the deletion program has edited it. The delete program can therefore only remove the contents of an archive file from the database once the archive file has been stored. The option offers enhanced backup, as the archived data is not deleted until it has been stored.

With storage before delete, you can use the checkbox *Deletion Program Reads from the Storage System* to control the read behavior of the deletion program during the deletion process.

- **Checkbox marked:** After storage, the stored archive file is deleted from the file system. Consequently, during the deletion phase, the delete program reads the archive file from the storage system. This ensures that the data is only deleted from the database if it was successfully stored in the storage system.

- **Checkbox not marked:** After storage, the archive file is not deleted from the file system. This setting results in improved performance of the delete program without neglecting backup considerations.

If, in the settings for the delete program, the radio button *Start Automatically* is also selected, the delete program is started automatically following storage. It therefore makes no difference whether the delete program is run in test or live mode.

Similarly, if you select the radio button *After Event*, deletion events are scheduled, but they are not run until triggered by the event that has been set up in the settings for the delete program.

Settings for Delete Program

Use

Here, you specify several parameters for the delete program.

Features

- You can set up a *test run variant* by modifying the parameters accordingly. A test run variant does not delete data from the database.
- You can set up a *live run variant* by modifying the parameters accordingly. A live run variant deletes the archived data from the database.



Delete program variants are client-specific. Consequently, you must make the variant settings in all the clients for which you want to carry out archiving sessions.

When archiving for the first time using a selected archiving object, the relevant test run and live run variants for the delete program must be fully maintained. Check whether both variants exist. Note especially, that if a live variant is not available, data is not deleted from the database when the delete program is run.

As a part of the delete job, you specify when the delete program is to be run in the background.

- If you activate the *Not Scheduled* radio button, the delete program is not run automatically. Instead, you can run the program at any time after archiving.
- If you activate the *Start Automatic* radio button, the delete program is run automatically immediately after archiving. However, if it was specified in Customizing that the file storage is to take place before the delete phase, the delete program is not started until the file has been stored. See also [Archive File Size](#).

This radio button is of interest primarily if you want to save the archive files before the delete run. If there is a long time period between the write run and the delete run, there is a danger that the data could be changed before deletion, which means that the data in the archive and in the database are no longer the same. The delete run should therefore be carried out as soon as possible after the write run.



Since *Before Images* has been deactivated for the rollback segments, there ought to be enough space available (Oracle's Rollback Tablespace).

- If you activate the *After event* radio button, the delete program is run automatically once a specific event has occurred. You select the relevant event by using the input help for the *Event* field. If an event requires that a parameter be set (an argument), enter the required parameter in this field. A parameter qualifies an event. See also  [Background event](#).

After the completion of all delete jobs for an archiving session, the ADK issues the event SAP_ARCHIVING_DELETE_FINISHED. The event number is the number of the archiving session. You can use this event for event-driven scheduling of accumulated archiving jobs. and the automatic backup of archive files using external tools.



You can use transaction *SM62* to display and maintain events.

- You can use the *Build index* indicator to specify whether an index is to be constructed for a particular archiving object. The index enables you to access specific data objects from the archive file.



We recommend that you use the single document access function in the [Archive Information System \(SAP AS\)](#).

Customizing: Postprocessing Settings

Use

Here you can schedule a postprocessing program to run after a delete run.

Prerequisites

The archiving object being used has a postprocessing program.

Features

- Set the *Run automat.* flag to perform *Postprocessing* automatically after deletion.
- Specify the postprocessing program variant in *Variant*.



The postprocessing program only runs when:

- The last delete program in the archiving session is finished
- No archive files have the status *Archived*, *Archiving* or *Deleting*.

Server Selection

Use

Server Selection allows you to define on which server(s) the archiving and delete programs should run as background programs.

Prerequisites

There must be active background processes running on the selected servers. Otherwise, the archiving or delete program will run on the database server (if a background process is available there) or on other selected application servers with active background processes.

Features

You can create an individual server group for each archiving object or you can select a single server group for all archiving objects.

You should decide when you save the selected server group:
Save for Object means that you create a server group for a selected archiving object, *Save for All Objects* means that you create a server group for all archiving objects.

If you do not select a server, or if you accidentally select a server on which there are no active background processes, then the archiving program will run on the database server if a background process is available there (network load reduction). Otherwise, both the archiving program and the delete programs will be scheduled randomly on application servers where active background processes exist.

If the database server has at least one active background process and is within the server group, the archiving program will automatically be scheduled on it to reduce network load.



A reduction in the number of application servers can lead to an increase in the overall runtime of the whole archiving process, especially where there are less active background work processes than delete programs running.

Only select servers on which there are active background work processes during archiving.

Activities

If you have created a server group, the system will also start other archiving-related activities within that server group, such as reloading, reading from archive files and constructing/removing indexes.