Introduction to SAP Data Archiving

Overview of the technology and functions

Version 1.1

May 2005
Copyright

© Copyright 2005 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft®, WINDOWS®, NT®, EXCEL®, Word®, PowerPoint® and SQL Server® are registered trademarks of Microsoft Corporation.

IBM®, DB2®, OS/2®, DB2/6000®, Parallel Sysplex®, MVS/ESA®, RS/6000®, AIX®, S/390®, AS/400®, OS/390®, and OS/400® are registered trademarks of IBM Corporation.

ORACLE® is a registered trademark of ORACLE Corporation.

INFORMIX®-OnLine for SAP and INFORMIX® Dynamic Server™ are registered trademarks of Informix Software Incorporated.

UNIX®, X/Open®, OSF/1®, and Motif® are registered trademarks of the Open Group.

Citrix®, the Citrix logo, ICA®, Program Neighborhood®, MetaFrame®, WinFrame®, VideoFrame®, MultiWin® and other Citrix product names referenced herein are trademarks of Citrix Systems, Inc.

HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

JAVA® is a registered trademark of Sun Microsystems, Inc.

JAVASCRIPT® is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, SAP Logo, R/2, RIVA, R/3, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP Business Suite Logo and mySAP Business Suite are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

Symbols used

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Exclamation Mark]</td>
<td>Caution</td>
</tr>
<tr>
<td>![Arrow Up]</td>
<td>Example</td>
</tr>
<tr>
<td>![Arrow Right]</td>
<td>Note</td>
</tr>
<tr>
<td>![Box]</td>
<td>Background information</td>
</tr>
</tbody>
</table>
Table of Contents

INTRODUCTION ........................................................................................................................................... 6

CONTENTS AND OBJECTIVE OF THIS GUIDE ............................................................................................ 6

1 THE PURPOSE OF DATA ARCHIVING ........................................................................................................... 7

1.1 THE CRITICAL EFFECTS OF GROWING DATA VOLUMES ....................................................................... 7

1.2 WHAT ARE THE BENEFITS OF DATA ARCHIVING? ................................................................................. 8

1.2.1 Greater System Availability .................................................................................................................... 8

1.2.2 Improved Performance and Response Times ........................................................................................... 8

1.2.3 Save Costs by Optimizing Your Available Resources ............................................................................ 8

1.3 DATA ARCHIVING IN THE CONTEXT OF INFORMATION STORAGE ........................................................... 9

1.3.1 Requirements for Stored Data ............................................................................................................... 9

1.3.2 Print Lists ............................................................................................................................................. 11

1.3.3 Data Retention Tool .............................................................................................................................. 12

1.4 DATA MANAGEMENT: PREVENTION, AGGREGATION, DELETION, AND ARCHIVING ........................... 12

1.5 PURPOSE AND SUITABILITY OF DATA ARCHIVING .............................................................................. 14

2 THE BASIC DATA ARCHIVING FUNCTIONS ............................................................................................. 16

2.1 BASIC TERMS USED IN DATA ARCHIVING .......................................................................................... 16

2.1.1 Archive Development Kit ....................................................................................................................... 16

2.1.2 The Archiving Object ............................................................................................................................. 17

2.1.3 Archive Administration .......................................................................................................................... 18

2.1.4 Archiving Session .................................................................................................................................. 18

2.1.5 Archive Management ............................................................................................................................. 19

2.1.6 Residence Time ...................................................................................................................................... 19

2.2 THE DATA ARCHIVING PROCESS ........................................................................................................... 20

2.2.1 Data is Written to the Archive ................................................................................................................. 20

2.2.2 Deleting Data from the Database ............................................................................................................. 22

2.2.3 The Storing of Archive Files .................................................................................................................. 23

2.2.4 Other Processes and Functions ............................................................................................................... 24

2.3 IMPORTANT DATA ARCHIVING FEATURES ......................................................................................... 26

2.3.1 Data Security ........................................................................................................................................ 26

2.3.2 Archiving in Online Mode ..................................................................................................................... 26

2.3.3 Data Compression ................................................................................................................................ 26

2.3.4 Release- and Platform-Independent ...................................................................................................... 26

3 TECHNOLOGY AND ADMINISTRATION .................................................................................................. 27

3.1 THE BASIS TECHNOLOGY OF THE SAP ARCHIVING SOLUTIONS: ADK ............................................ 27

3.1.1 ADK Positioning and Components ......................................................................................................... 27

3.1.2 ADK Runtime Environment .................................................................................................................. 27

3.1.3 ADK as a Development Environment ................................................................................................... 29

3.2 TASKS OF THE DATA ARCHIVING ADMINISTRATOR ........................................................................... 29

3.2.1 The role “Data Archiving Administrator” ............................................................................................... 29

3.2.2 Analysis ................................................................................................................................................. 30

3.2.3 Monitoring ........................................................................................................................................... 32

3.2.4 Settings ................................................................................................................................................ 34

3.2.5 Data Archiving Statistics ......................................................................................................................... 36

3.2.6 Reorganization of the Database After Data Archiving ......................................................................... 37

3.3 AUTOMATED PRODUCTION OPERATION .............................................................................................. 39

3.3.1 Periodic Archiving ................................................................................................................................. 39

3.3.2 Scheduling Data Archiving Jobs .............................................................................................................. 40

3.3.3 Interrupting and Continuing the Write Phase ......................................................................................... 41

3.3.4 Options for Automating Dependent Processes ..................................................................................... 42

3.3.5 Controlling Data Archiving Jobs Using an External Job Scheduler ....................................................... 42

4 STORING ARCHIVED DATA ...................................................................................................................... 43

4.1 INTRODUCTION ........................................................................................................................................ 43

4.2 STORAGE ON AN SAP CERTIFIED STORAGE SYSTEM USING ARCHIVELINK ................................. 44
5 ACCESS TO ARCHIVED DATA........................................................................................................52

5.1 INTRODUCTION .......................................................................................................................52
5.2 THE FUNDAMENTALS ...............................................................................................................53
5.3 SEQUENTIAL READ PROGRAMS ............................................................................................53
5.4 ARCHIVE INFORMATION SYSTEM ..........................................................................................54
  5.4.1 Creating an Infostructure ....................................................................................................55
  5.4.2 Activating an Infostructure ...............................................................................................55
  5.4.3 Building an Infostructure ................................................................................................56
  5.4.4 Displaying an Infostructure ..............................................................................................56
  5.4.5 Deleting an Infostructure ................................................................................................57
  5.4.6 Using Display Variants .....................................................................................................57
5.5 DOCUMENT RELATIONSHIP BROWSER .............................................................................57
  5.5.1 DRB and Data Archiving ...................................................................................................57
  5.5.2 Searching Options ..........................................................................................................57
  5.5.3 Configuring and Personalizing DRB ................................................................................58

A. GLOSSARY .................................................................................................................................60
B. ADDITIONAL INFORMATION AND SERVICES.......................................................................63
  SAP Service Marketplace .............................................................................................................63
  SAP Library .................................................................................................................................63
  Archiving Your SAP Data ............................................................................................................63
  Training Courses ........................................................................................................................64
  Consulting and Services ..............................................................................................................64
**Document History**

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>May 2004</td>
<td>n.a.</td>
</tr>
<tr>
<td>1.1</td>
<td>May 2005</td>
<td>Corrected minor translation errors</td>
</tr>
</tbody>
</table>
Introduction

Even the most modern and technologically advanced database systems can suffer from performance bottlenecks caused by large data volumes. On the application side these bottlenecks manifest themselves in the form of poor system performance and on the administration side in the form of an increased use of resources. High data volumes can also have a considerable effect on the Total Cost of Ownership of a system, in spite of falling storage prices.

To avoid the negative effects of large data volumes on costs, performance and system availability, business complete data, which is data no longer needed in everyday business processes, should be removed from the database. However, simply deleting this data is not a useful option in most cases, because often times the data still needs to be available for read accesses. Therefore the data needs to be removed from the database and stored in such a way that it can still be read-accessed later.

SAP Data Archiving is the only method supported by SAP to remove application data from the database in a consistent, secure and comprehensive manner. Consistency is ensured through the use of checks performed by the archiving programs. A purely database-integrated archiving is not used, because the database does not know the business context of the data to be archived. Using data archiving you can select significant objects, such as accounting documents, material master records or HR master data, and remove them from the database, without having to worry about the fundamental table design of the linked data.

The archived data is stored in a file system and from there can be moved to other storage media. For security reasons the archived data is not deleted from the database until the archive files have been read and hereby confirmed.

Contents and Objective of this Guide

This guide is meant to provide you with an introduction to SAP Data Archiving. It describes the technical foundation of data archiving and outlines the relevant functions and processes. The guide addresses all those, whose job includes data archiving tasks, such as IT administrators and members of a project team in charge of archiving SAP application data. The aim of this guide is to provide readers with current and concrete information, so that they can quickly familiarize themselves with the topic and perform the data archiving tasks they are responsible for.

The guide is not meant to provide a detailed view of each data archiving function. This would go beyond the scope of this guide. For detailed information about the technical processes and individual archiving objects (see Chapter 2.1.2), see the SAP Library, the SAP Data Archiving Website or our book, "Archiving Your SAP Data". For more information about these sources of information, see the appendix.

This guide contains the following chapters:

- Chapter 1 describes why it is important to archive application data and mentions some important aspects to be considered. It also explains how data archiving is positioned by SAP and the importance of data management in this context.
- Chapter 2 describes the basic data archiving functions and processes and gives an overview of SAP Data Archiving.
- Chapter 3 describes the technical aspects of data archiving. It describes the technology involved and explains how an administrator can integrate data archiving efficiently in his or her current system processes.
- Chapter 4 covers the storage of archived data. It gives an overview of the different concepts and the available technology and offers guidelines as to which storage strategy a company can follow based on its requirements.
- Chapter 5 describes the different options available for accessing archived data. The main focus of this chapter is the Archive Information System and the Document Relationship Browser.
- The appendix contains a glossary of the most important terms in data archiving and offers a list of further sources of information on data archiving.

The archiving functions and screenshots used in this guide are taken from SAP R/3 Enterprise 4.70. They should be viewed as examples of data archiving in an SAP System and may vary in appearance depending on the mySAP Business Suite components you are using.
1 The Purpose of Data Archiving

1.1 The Critical Effects of Growing Data Volumes

Data that belongs to closed business processes very rarely needs to be modified. Experience has shown that in financial accounting, for example, documents from closed posting periods are only accessed in exceptional cases. If these documents were to remain in the database, the data volume in the database would grow continuously. Depending on the growth rate, it would be only a question of time until the database management system reached its technological limits, or until you would encounter other problems.

Figure 1 shows this relationship using two different scenarios. In scenario 1, represented by the dotted line, data archiving is not being used. As a result the data volume in the database rises continuously and most likely will reach a critical limit relatively quickly. Scenario 2 is represented by the solid line. Its step-like form is the result of regular archiving sessions. Immediately after archiving the data volume recedes noticeably and then begins to rise again. As a result the database grows overall more slowly than it would without data archiving. This has many advantages, which we describe below.

Figure 1: Database Growth with and without Data Archiving

This example is typical for many business systems: The database contains data of closed business processes, although this data is no longer needed in everyday business processes and should no longer reside in the database.

To avoid bottlenecks in your system, which are mainly caused by the amount of data in the database, you have the following options:

- **Expand your system resources**
  
  This implies upgrading your hardware, which would entail for example adding more disk space, increasing your main memory, or improving your CPU. However, it is only a matter of time before your system’s limits would be reached and performance would be affected again. Moreover, upgrades of this kind usually involve high costs. If you are facing especially strong data growth, data archiving would be preferable, because it offers a better cost-benefits relationship than a hardware upgrade does.

- **Reduce or limit data growth**

  The goal of this strategy is for you to be able to live with your existing resources for as long as possible, without having to perform constant hardware upgrades. The idea is to focus on your current data volumes.
in the database and data growth. By actively working on these parameters you will be able to most optimally use your system resources. You can reduce data volumes by deleting or archiving data, if it makes sense from a business point of view. You can also use data prevention in addition to the aforementioned measures to avoid the creation and persistent storing of unnecessary data, such as log files or spool requests. For more detailed information see the section entitled "Data Management: Prevention, Aggregation, Deletion, and Archiving".

1.2 What Are the Benefits of Data Archiving?

Large data volumes do not only affect the performance and manageability of your database. They also influence other factors, which are described in more detail in the following section.

1.2.1 Greater System Availability

Large data volumes can be the cause of long runtimes during regular administration tasks on your system, such as data backups. This is especially detrimental during operations that require that an application system or certain parts of a system be shut down, meaning that during that time the system is not available to the end user. A shut down may be required during an upgrade to a higher software release or after a system failure, when the data is being restored. Thus, data archiving can help minimize the time a system is unavailable, by reducing the volume of data in your database. In addition, data archiving can take place while your system is online, which means you do not have to shut down your system during archiving operations.

1.2.2 Improved Performance and Response Times

Large data volumes affect the performance of your database and application server. The increased amount of data in the application tables means that during application-specific tasks, especially reporting, but also transactions, there is more data to process for the selection. This in turn generates a greater memory load and a greater burden on CPU performance on your servers. To improve the performance of your hardware, it is often necessary to expand your hardware resources. However, there is a limit, technically speaking, as to how much you can expand your hardware. For example, it is not possible to continuously increase the database buffer, due to the limitations of the database architecture.

The response times of your system can also be highly affected by large data volumes in your database. Not only do high data volumes mean that users have to wait longer for their query to be returned, but also, that unspecific searches can lead to the transfer of large amounts of data, which may not be needed.

With large data volumes you can only achieve short response times by using increased hardware resources. Therefore, any queries that would be performed online with small data volumes may have to be run in the background instead. In general with large data volumes we can say that queries involve data that is actually no longer needed in everyday business operations, because it belongs to closed business processes. For indexed searches, this means large index trees. The processing time here is proportional to the searched data volume.

Because the amount of data that can be selected is larger with a greater data volume, the end user will have more data returned for his or her query. This makes it especially difficult for the end user to pick out the data he or she was actually looking for. It may be necessary for the end user to restrict the data selection more, which would require him or her to make additional user entries.

In sum, we can say that data archiving helps ensure that only data still needed in everyday business is kept in the database. This in turn prevents a slow deterioration of your system performance and response times, caused by large data volumes. In addition data archiving generates benefits during data analysis. With less data to be searched information can be located more quickly.

1.2.3 Save Costs by Optimizing Your Available Resources

The memory area where application data is stored is the safest, technologically most advanced and therefore also the most expensive storage area within a company's system. In addition, storage systems are often mirrored, meaning that the data on hard disks is often redundant.

When database systems are growing rapidly, worries often center around the storage system. If however, you also take into account how large data volumes affect system performance, then there are other
important performance parameters and resources that come into play, such as the number of processors and their speed, as well as the size of the main memory. In an extreme case scenario the database server will have already reached its load limits and cannot be expanded any further. In this situation a company would have to look towards other options that may involve new investments.

There are also other factors that affect the overall cost of a company's system. These include the fact that data needs to be transferred from one production system to other development and consolidation systems. As a result it is not uncommon for a company to have three to four copies of its entire data set. The pure costs of the storage disks then only comprise a small fraction of total storage costs; administration costs are five to seven times higher than the cost of hard disks, and including the costs of database administrators often times lie even higher. If we take all these factors into account, it is not surprising that customers are paying up to several thousand euros per gigabyte of production disk space. Thus, any reduction in your total data holdings will help you use your existing resources, such as hard disks, memory, CPU, and network, and put off new investments for a longer period of time. Lower data volumes also mean that you will have to spend less time on database administration, which of course has a positive effect on overall costs. From these arguments we can see that data archiving plays an important part in lowering a company's TCO and helps generate a faster ROI.

1.3 Data Archiving in the Context of Information Storage

As mentioned, data archiving is used to remove data that is no longer needed in everyday business processes from the database, in order to help you optimize your database resources. However, it is not enough to view data archiving merely as a means to optimize resources and improve the administratability of your system. Data archiving is also of utmost importance in the context of data retention requirements and legal tax compliance.

The primary purpose of storing information is to facilitate read accesses to this data in the future. You can use data archiving to store data, and also print lists. In practice both options are frequently used together. Often companies have not only implemented a system to handle its business processes, such as SAP R/3 Enterprise, but also other components, such as SAP Business Information Warehouse (SAP BW) or other front- or back-end systems that are connected via standardized interfaces.

1.3.1 Requirements for Stored Data

1.3.1.1 Technical Requirements

From a technical point of view it is important that the archived data can be read accessed in the future, regardless of which hardware was in use when the data was archived or the release status of the software. To make this possible additional information is stored together with the archived application data, to indicate with which hardware the data was written to the archive and which data structure was used. For more detailed information about this process see Chapter 3.

From a technical standpoint, data archiving must meet the following criteria:

- **Transformation**
  
  When data is removed from the database it is important that the relevant information is transferred correctly from the database to the corresponding storage medium. This includes a possible change in the storage medium, in case of technological upgrades, for example. However, it is possible for technical errors to occur during the transfer of data, and one has to be aware that such processes are always marked by a certain amount of risk. Therefore, authorities can request that companies maintain automatically or manually generated logs that confirm that the information contained in the archive files is identical to the information contained in the database. For security aspects in data archiving see Chapter 2.3.1.

- **Storage**
  
  The chosen tertiary medium must guarantee that for as long as the data is being stored it is preserved, both in terms of its visual composition and its content. During the time of storage the data being kept on the storage medium, such as a magnetic tape, can be affected by physical and chemical processes. A storage medium must therefore have the ability to maintain the data that is being stored on it to ensure that the data can still be called up over time. If according to the manufacturer of the storage medium you have to adhere to an “expiration” date due to natural causes, make sure you move your data to another medium before the date.
• Display
The archiving solution must ensure that when archived data is accessed it can be read and interpreted, and that it is possible to display the content of the data in a visually correct manner. In this context readable means that the information can be displayed by a third person using the appropriate read programs.

1.3.1.2 Business Requirements
From a business point of view only data that is no longer needed in everyday business processes can be removed from the database. Special checks have to be performed to make sure that only data from closed business processes is archived. Business objects within a business process can have many dependencies, due to the high integration of applications in the SAP System. Therefore, before archiving a specific object, checks have to be performed in case other objects have to be archived first or at the same time as the business object in question.

Although data archiving can also be used for master data, such as material, condition or HR data, transaction data makes up the bulk of the data that is archived. Master data generally remains in the system longer and uses less space in the database than transaction data. Therefore master data is not archived as often as transaction data. Transaction data is created during practically every transaction within a process chain, which causes data growth in the database. Therefore transaction data is also called mass data.

Figure 2 shows a part of a business process used to complete a sales order in the SAP R/3 Enterprise application Sales and Distribution (SD) and the corresponding archiving objects. A delivery is created based on a sales document. An invoice is then created in response to the delivery, and the sales process in SD is completed. The invoice is then sent to financial accounting, where more documents are created.

![Diagram of a business process](image)

**Figure 2: Documents and Archiving Objects Used During the Business Process "Sales Order"**

Depending on the complexity of the business processes other documents could be created in addition to the ones generated in SD. This could be in the area of Production Planning (PP), if goods are produced internally. Unlike most master data the documents that were created as part of this kind of process chain are only relevant to operations for a limited amount of time. They can therefore be removed from the database after a relatively short period of time, using the appropriate archiving objects.

An example of master data that is often archived is product data. Depending on the life cycle of the product, the corresponding data may have to be kept for several decades. The customer service departments of a company have to be able to access certain product information during the entire life cycle of the product. This includes construction relevant CAD drawings, bills of material, built in components, production resource tools, and task lists. This list shows that in these processes the system has to access different data objects that characterize the same final product. In certain cases it may even be necessary to reconstruct the entire process in which the product was involved.

During internal revisions it is often necessary to access archived data, to monitor specifications and provide material to support decisions. Generally internal revisions take on the monitoring and control functions of the management. Here it is important for the archiving solution to guarantee truth and fairness, security, and efficiency. Internal revisions can touch any area of a business. Therefore, it is important to decide within a company which data should be kept in addition to that required by law.
Statistical analyses bring together important information about production, sales, and markets, and help the management make decisions for their business. Some analysis programs offer you the option to also analyze, format and weight archived data. However, this is not very common, because for technical reasons, analyzing archived data can involve long runtimes. More specifically, this has to do with how the analysis program selects the archived data. An analysis involving archived data requires the system to open many archive files and read all of the relevant data objects, in order to select the requested fields. As a result runtimes for the analysis of archived data can be very long. Therefore, if you need to run comprehensive analyses that include archived data, we recommend that you use a data warehouse, such as SAP BW.

1.3.1.3 Legal and Tax Requirements

Businesses have to follow general accounting rules and produce legally compliant, complete, correct and verifiable data. Although this information may vary from country to country in terms of its content, all countries adhere to the same general accounting guidelines. Therefore, we will not discuss these any further in the context of data archiving. We want to focus on tax relevant data in this section. Considerations in this context are becoming increasingly more important, because many countries are switching to paperless tax audits, that is tax audits take place on electronic data instead. For example, the new German law that went into effect on January 1, 2002 in Germany (GDPdU), sets forth regulations regarding the actual data being audited, but also the form in which the auditor has access to this data.

Although tax laws may vary greatly from country to country, data retention requirements seem to be a common denominator among them. These requirements determine that tax relevant documents may not be deleted. They must be kept in case they are needed during a tax audit. However, it is important to differentiate between ‘keeping or storing data’ and ‘data archiving’. How the data is to be ‘kept’ and for how long can differ from country to country. Therefore it is difficult to make any general statements in this respect.

In Germany, for example, a company is required to keep most of its documents for ten years, although some only have to be kept for six. Another common requirement is that data must be kept in such a way that it cannot be changed, to avoid any type of data manipulation.

What role does data archiving play in this context? Data archiving was not designed to be used for tax purposes or to meet other requirements in the context of data storage. Rather its purpose was to help reduce the data load on the database. Data archiving deals with data from completed business processes, which only needs to be accessed in the read-only mode. Therefore, the display programs for archived data often do not have the same functions as display programs for online data that resides in the database. Data archiving ensures that no data is lost. The data is merely moved from the database to the file system and continues to be part of the original system in which it was produced. It has the same significance as the database data, which has not been archived yet. This means that tax authorities have access to all data in the system, with some more restricted reading functionalities for certain data.

If, however, the data is taken out of the original system, for example by being stored on a storage system, which cannot be accessed from the original system, then it is necessary to also archive the context of this data, to ensure that it can be interpreted at a later point in time. The context must be limited to only the most necessary information, due to storage space limitations. SAP offers a tool called DART to extract tax relevant data from the original system. It has functions that can be used to create transparent files with actual data, and to display this data. For more detailed information about DART see Chapter 1.3.3.

Finally, we would like to refer back to the differences that exist between the different countries. Because of these differences it is impossible for us to issue any general statements as to which and how much information must be stored and for how long. It is very important that you familiarize yourself with the requirements specific to your country. For more information about data archiving in the context of GDPdU and DART see SAP Note 670447.

1.3.2 Print Lists

Print lists are the result of SAP reports. An SAP system is delivered with a number of predefined reports, such as accumulated balance audit trails, warehouse inventories, balance sheet valuations, journals, etc. Print lists can be printed on paper or stored on a storage system via ArchiveLink. Electronic storage of print lists has several advantages: It saves space, data can be made available more readily because it is stored

---

1 In this case we are not referring to the storage of archive files on an external storage system connected via the ArchiveLink interface, because with ArchiveLink storage systems the data can still be accessed from the original system. Rather we are referring to cases where alternative archiving and storage processes, which are different from SAP data archiving, are used.
centrally, it is possible to conduct indexed or text-based searches, and the reports, which are often used for
tax audits, can be managed centrally. Print lists are produced as long as the data is in the database. A
stored or printed print list represents a defined business data relationship, which is based on the SAP data
set as it appears at the time the list is stored or printed.

Because print lists can be created individually at any time, existing print lists must differentiate whether they
are based on already stored data or data that still resides in the database. As of SAP R/3 3.1 it is possible
to refer to already stored documents from the print lists themselves, using hyperlinks. If, for example, the
document number "4711" is part of the print list information, you can set a hyperlink to the already scanned
or stored original document invoice "R816". Depending on the type of print list used, the actual data
relationship in the SAP system, which implies a new version of the print list, can vary from the original print
list. It is also possible that the data of the stored original document no longer coincides with the data of the
SAP document. This is especially important to remember with print lists that are used in tax audits.

Print lists are not used to facilitate data searches. Rather they provide an excerpt of the data in a system at a
specific point in time and in a specific business context. Unlike other data that is archived, print lists do not
need context information to be interpreted, because they already contain the context information. For this
reason, print lists can be viewed as complete information units that can also be interpreted outside the
original SAP system.

As of SAP R/3 3.1 it is possible to conduct indexed searches using freely defined indexes within a print list.
Free searches in the list area currently being read has always been possible. The Archive Information
System offers functions for searching historical, meaning archived data. ArchiveLink offers a search function
for stored print lists. You can search via the management record using the object type, document type and
the report name of the print list. In addition, the Document Management System (DMS) offers the function to
access print lists via individually definable search criteria; also, ArchiveLink can store document information
records in DMS.

1.3.3 Data Retention Tool

The Data Retention Tool (DART) was originally developed to meet the specific requirements of US tax
authorities regarding data storage and the analysis of data. It was mainly implemented in the North and
South American markets. As of SAP R/3 4.5 the program is delivered as part of the Standard. Today SAP
offers new functions to meet the requirements of the German tax regulations (GDPdU). The new functions
are available as of DART 2.0. For example, you can use DART to extract additional data segments that are
relevant for German tax purposes. The current version out is DART 2.2 and is valid for all Release tracks as
of SAP R/3 3.1I. For more information about this topic see SAP Note 668943.

DART is used to extract tax-relevant data from the SAP R/3 or SAP R/3 Enterprise system and to store this
data in an external file. To do this the program first reads the business objects and then filters the relevant
field contents out of these objects. Then the data is stored in temporary database tables, which are not,
however, identical with the actual tables that describe the physical data model of a business object. The data
is stored sequentially in an uncompressed clear text format and can be analyzed with external tools. It is also
possible to display this data extract using DART.

1.4 Data Management: Prevention, Aggregation, Deletion, and Archiving

Although data archiving is a very effective tool for reducing large data volumes, it should not be the only
measure used to battle data growth. Generally speaking, data archiving should only pertain to data that was
actually used previously in the system. For data that is not necessary, there are other alternative measures
you can use to stop your database from growing. It always makes sense to first check whether it is actually
necessary to update certain data, or whether specific groups of data can be aggregated. Logically data
archiving, too, can benefit from smaller amounts of data.

We recommend that in order to most effectively reduce your data volumes you apply the following four-step
method in the listed order, before you start archiving or at the latest during your archiving project:

1. Data Prevention

For certain data it is technically possible to switch off updating. If from a business point of view this data is
not necessary, you should switch off the updating of the data. For example, in spool management you
indicate that spools should be automatically deleted after their output. This will considerably reduce
the growth of spool table TST03 in the future.
2. Aggregation

In some cases it is possible to aggregate data, for example through forming sums. If aggregated information fits your requirements for certain data, you should make use of this method. An example of how to apply this method would be line item aggregation in CO. As a result you avoid writing a line in CO for every line that is written in the original document (for example a material booking). Note that aggregation does not have an immediate effect, because it only influences postings that take place in the future. Old documents are not affected, and may require archiving.

3. Deletion

A lot of the data that is not required to be stored in an archive can be deleted from the system shortly after its creation. An example of this type of data are batch input folders. Generally, these folders are created when data is transferred to the SAP system in the background, for example during a legacy data transfer. These folders are not needed after the transfer. Processed folders can then be deleted using a delete program.

4. Archiving

Archiving should be used for data that cannot be prevented nor deleted. Check how long your data needs to remain in the system from a business point of view. Only archive the data that is no longer needed in the processes of the production system. Archiving is not a cure-all means to reduce your data volumes as much as you like.

The following figure can help you decide when to use data archiving:

![Decision Tree]

**Figure 3: Deciding If and When to Archive**

Data archiving should be an integral part of every company’s data management process, as should the other methods for reducing data mentioned in this section. For more information about data management, including detailed recommendations for specific critical tables see the Data Management Guide, which you can download from the SAP Service Marketplace (/data-archiving → Media Library → Literature & Brochures).
1.5 Purpose and Suitability of Data Archiving

The following section answers some frequently asked questions about data archiving, how SAP positions it and when and for which purpose it is best implemented. In this section we also want to address what data archiving is not suited for.

1. What is the purpose of data archiving?

Archiving application data is an important part of managing mass data. Unlike backup and recovery, data archiving is a process that stands in close relationship with the applications and directly affects the business processes of a company.

Business solutions such as SAP R/3 Enterprise are designed to save data to the database and access this data when necessary. When you archive this data it is still available to the applications for display purposes, but can no longer be changed. Read access to archived data may be restricted compared to the access to database data. However, read accesses are possible for all important transactions. To develop a similar functionality for archived data as that used for online data would be very difficult, because the archived data would constantly have to be adapted to changes in the system, such as conversions. The benefits from an online-like display functionality for archived data would not justify the effort needed to develop such a functionality, since in reality archived data is not accessed very often. For this reason SAP offers alternative tools for accessing archived data, such as the Archive Information System.

2. Which data should be archived?

This question is best answered by asking which data should not be archived. Storing application data in archives is not the equivalent to having data in the database. Applications are designed for accesses to the database and therefore all data that is needed for operation by the applications belongs in the database. If, for example, for a rebate settlement you need to process all the invoices of a given business year, the corresponding data can only be archived once the rebate settlement has been completed.

3. When should application data be archived?

Application data should only be archived when the following criteria has been fulfilled:

- It is no longer needed in any transactions or processes, such as for completing the annual balance sheet.
- It does not have to be changed anymore.
- The data probably does not have to be displayed very often anymore.

Before you start archiving make sure you fulfill all documentation requirements by creating all the necessary information (such as DART extracts) you may need for later audits.

4. Is data archiving enough for auditing purposes?

Data archiving was not designed as a tax and audit tool. It can support you in meeting the requirements of tax authorities, by conserving data and making it available over a longer period of time. The tax and audit tools are the Audit Information System (AIS) and DART. If you want to use DART you should do so before you archive your data.

During an audit it is also possible to access archived data, in case more detailed information is needed that does not appear in the other documents you previously created. Archived data should only be included in the auditing process if additional data is requested for the audit. To make the auditing process easier you should try to meet all requirements during the time the data is still in the database.

5. When is data archiving beneficial?

Archiving application data is beneficial when the effort spent on maintaining your database is becoming too expensive and when, at the same time, you can store and manage the archived data without spending huge amounts of money.

If the costs of accessing the archived data grow higher than the costs of storing and managing the data in the database, then the data should remain in the database. If you are weighing the costs of using data archiving, possibly to avoid having to purchase a new, more powerful database server, you should also include the costs of accessing the archived data in your calculation.
6. What status does archived data have?
Archived data cannot be modified and can therefore no longer be used in the processes of current business operations. It is inseparable from the system in which it was created and can only be accessed and interpreted from this system. If you need historical data for informational purposes, you can read access the archived data. However, because this data has been stored away from any changes in the system (for example a reorganization), it cannot be guaranteed that its contents and structure match the context currently being used. Moreover, some archiving objects only offer a very restricted access to archived data – for example only with single document displays or evaluations.

7. Can archived data be reloaded into the database?
From a technical standpoint, archived data can be reloaded into the database. However, because we are dealing with historical data, which has not been part of any changes in the database, you run the risk of generating inconsistencies in your database. Therefore we always discourage the reloading of data back into the database.

An exception would be if you reload archived data immediately after it has been archived. A reload of this kind would be an emergency measure after a failed or unsuccessful archiving session, because, for example, you archived the wrong data due to an error in the Customizing settings. Reloading is only possible with some archiving objects.

8. When should you start archiving your data?
You should begin with data archiving before it is too late and you have exhausted all the alternative measures for improving the condition of your system. This includes planning how big your system needs to be based on your anticipated data volume (sizing), and determining the residence times of your data. The latter point refers to the amount of time the data must remain in the database. You should also identify and fulfill any audit requirements before you begin archiving your data.

Always keep in mind that data archiving slows down the growth of your database. It cannot stop the growth completely. Therefore the goal of data archiving is to keep your system under control over a long period of time, not to return your system to a controllable state.

9. How can you access archived data?
There is not a general answer to this question valid for all components of the mySAP Business Suite. How archived data is accessed depends on the application and the archiving object that was used. You have the following options to access archived data:

- Ideally the user displays the archived data directly from his or her usual transaction. However, this can only work if index information about the archived data is kept in the database. This can be done using the Archive Information System (AS), for example.

- In certain cases it is even possible to run an analysis using archived data. Because these types of evaluations usually involve long runtimes (see 0) and usually do not make sense from a business point of view (because the data stems from closed business processes), only a few reports are available for this purpose. Therefore, if you need to run comprehensive analyses that include archived data, we recommend that you use a data warehouse, such as SAP BW.

An analysis function always depends on the application, which decides how detailed the result of the analysis will be and whether or not a mixed (online and archived data together) analysis is allowed. When data is displayed in a list form, you are not notified that the data selection may be incomplete, because some of the data has already been archived.

- The Archive Development Kit (ADK) and the Archive Information System (AS) can be used to adapt the access to archived data to customer-specific needs. For more information about how to do this, visit the training course BIT 670 or see the corresponding documentation.

10. Can print list archiving be used as a substitute for data archiving?
No, print list archiving is not a substitute for data archiving.

Print list archiving and data archiving complement each other. Print lists are produced as long as the data is in the database. Later print lists can be archived, by being moved to a storage system that is integrated via an ArchiveLink interface. Print lists are mainly created and archived to use in future...
evaluations and audits. If at a later point in time you find that you need more information for an evaluation or an audit, you can access the archived data at any time.

2 The Basic Data Archiving Functions

To better understand data archiving it is important to understand the technical basis of the functions and to get an overview of the relevant processes and the scope of the functions. The objective of this chapter is to introduce the most important terms used in data archiving and to outline the steps involved in the basic archiving process. At the end of the chapter we describe the most important features of data archiving.

2.1 Basic Terms Used in Data Archiving

The following are the most common terms in data archiving and need to be understood before delving further into the concepts of data archiving. For additional data archiving terms see the glossary.

2.1.1 Archive Development Kit

The Archive Development Kit (ADK) is the technology at the core of data archiving. It is responsible for controlling and managing the archiving sessions and makes sure that the data remains interpretable in the long term. ADK is also a programming interface for application programs and it is responsible for handling the archive files. ADK is used to create, open, write, read and close the archive files.

Runtime and development environment

ADK controls the archiving process within the runtime environment that it itself creates. It also provides a development environment for applications. SAP development can use ADK to develop new standard archiving objects and adapt existing archiving objects. Archiving customers can develop their own archiving solutions for their customer-specific tables. The following figure shows the integration of ADK in the archiving process and names its most important functions.

Figure 4: ADK Integration and Functions

For more detailed information on ADK see Chapter 3 “Technology and Administration”.

© SAP AG 2004
2.1.2 The Archiving Object

Another central component of data archiving is the archiving object. It determines which tables are accessed during archiving to make sure that a business object, such as an invoice or a financial accounting document, can be completely removed from the database. In addition, an archiving object also contains the archiving programs and Customizing settings, necessary for archiving its corresponding business object type.

More concretely, an archiving object has the following components:

- **Data declaration**
  Is used to define which data from the database makes up a business object.

- **Archiving programs**
  - Write program: Writes the business objects sequentially to the archive files.
  - Delete program: Deletes the business objects from the database after it has read and confirmed them in the archive file.
  - Preprocessing program (optional): Prepares the data for archiving. This includes, for example, the setting of an archivability indicator (also called a deletion indicator).
  - Read program: Is used to display an archived business object.
  - Postprocessing program (optional): Is used to process data after archiving; for example to update statistic data or indexes.
  - Reload program (optional): Reloads the archived business objects into the database.

- **Customizing settings**
  Used to define the archiving-object-specific parameters for a specific archiving session. These settings depend on the archiving objects and may therefore vary.

The following figure shows the different components of an archiving object.

![Components of an Archiving Object](image)

**Figure 5: Components of an Archiving Object**

- **Archiving classes**
  Another common term used in data archiving is "Archiving Classes". It refers to data objects that are not independently defined from a business process point of view, but which belong together from a technical point of view; these include SAPscript texts, change documents, and classification data. These data objects are created when a business object is created or modified and they are usually archived together...
with their corresponding business objects. Access to this archived data usually also takes place using archiving classes.

Archiving classes are developed by SAP, but they may be used in customer-specific programs. However, it is only possible to use them in connection with an archiving object.

2.1.3 Archive Administration

Archive Administration (transaction SARA) is the main tool used for most user activities in data archiving, such as the scheduling of write and delete jobs, building and deleting the archive index, or storing or retrieving archive files. It is the central component used to control and monitor archiving sessions and it is a standard user interface used by all archiving objects in the SAP system. The interface will only vary in the amount of programs the corresponding archiving object offers. Archive Administration automatically receives all relevant data and information about the archiving processes from ADK.

In addition, Archive Administration offers the following functions to be used by the archiving administrator:

- Archiving Customizing
- Overview of the archiving jobs
- Assigning tables to archiving objects (transaction DB15)
- Data archiving statistics
- Archive Information System

Figure 6 shows the entry screen of Archive Administration, as it would look using the archiving object FI_DOCUMENT.

![Figure 6: Entry Screen of Archive Administration](image)

2.1.4 Archiving Session

An archiving session is an organizational unit of the archiving process. The unit comprises a write and a delete phase, and may also include an optional storage phase. It is also used to refer to the data set that was archived during the write phase. It encompasses all the archive files belonging to an archiving object that were created at a specific point in time through the execution of the write program. Each archiving session receives a unique ID in the form of the archiving session number, which can be used to access and display the session as a whole in archive management.

The status of the archiving session shows whether the session is complete, incomplete or contains errors. In an incomplete session the write program was completed successfully, but the delete program has not been
executed. An archiving session has the error status, if the write or delete programs had to be terminated due to an error.

2.1.5 Archive Management

Archive Administration should not be confused with Archive Management. Archive Management is part of Archive Administration and is used to manage and display the archiving sessions and archive files. ADK saves management data for the different archiving sessions and archive files in ADK database tables, to enable users to access archived data later on. Some of this data can be displayed in Archive Management, such as:
- The number, date, time, and status of an archiving session.
- All archive files that belong to an archiving session

It is also possible to display other details for an archive file, by double clicking on it.

Deletion of archive files

After a certain amount of time archive files may no longer be needed. This is the case, for example, if the data no longer needs to be kept from a legal standpoint, or if it is available in another format in which it can still be evaluated and analyzed. This includes print lists, microfiches, and statistics systems. The decision whether or not archived data can be deleted depends on several factors and can only be made on an individual basis.

Deleting an archiving session involves the following steps:
- Deletion of archive files on an operating system level
- Deletion of the management data in the SAP system

It is only possible to directly delete management entries through archiving. However, first you must set the To Be Archived indicator in the detail screen of the corresponding archiving session in archive management. The name of the relevant archiving object for this operation is BC_ARCHIVE. The archiving session started with this archiving object deletes all management data of all marked archiving sessions from the database. If necessary, the management data can be reloaded.

2.1.6 Residence Time

Besides business process related criteria, which depend on the application, residence time is the most important criterion used to determine the archivability of business objects. The residence time refers to the amount of time that has to pass before a business object is archivable. Residence time is measured in number of days and is entered in application-specific Customizing for a specific archiving object.

The calculation of the residence time depends on different application-specific criteria and starting points. For a document from a logistics application, for example, the starting point could be the time it is entered into the system, the change date, or the completion date. Figure 7 shows how the residence time is calculated for a sales document. The residence time begins either with the creation date of the document, or, if the document was changed before the residence time was reached, it begins with the last change date. The appropriate option can be selected in application-specific Customizing of the corresponding archiving object SD_VBAK.

![Figure 7: Calculating the Residence Time for Sales Documents](attachment:figure.png)
2.2 The Data Archiving Process

Data archiving is used to write business objects that have fulfilled their archivability criteria from the SAP system database to archive files. In the subsequent delete phase the archived data is removed from the database using the delete program. As a result, the data only exists once in the system: in the archive. The archive files can be stored for the long term on various different storage media, such as storage systems, HSM systems, or tertiary media such as WORMs, CD ROMs or magnetic tapes. This means that they are still available for analysis if needed.

Data archiving makes sure that application data that is no longer relevant for current business processes, which however must be kept for legal purposes, is removed from the database and stored on less costly storage media. As a result, the database load is noticeably reduced and system response times are considerably improved.

The following figure shows an overview of the data archiving process.

Figure 8: The Data Archiving Process

The data archiving process in the SAP system can be divided into the following steps:

1. **Data is written to the archive**
   The data to be archived is read from the database and written sequentially to newly created archive files.

2. **Data is deleted from the database**
   The delete program deletes the data from the database after it has been completely written to the archive files. To ensure the integrity of the archived data, the delete session is not started until the created archive files have been read and confirmed.

3. **Archive files are stored**
   The archive files that were created during the write phase can be moved to storage systems or to tertiary storage media. The storage phase is not part of the actual archiving process. Another option would be to start the storage phase before the deletion phase.

2.2.1 Data is Written to the Archive

The write phase is the first step in the archiving process. During the write phase the write program is executed. It can be scheduled using Archive Administration. To schedule the write program enter the archiving object you wish to use, choose a variant, or create a new one, and set the start date and spool parameter. With respect to the variant, make sure that it is not being used in another write session. If the
same variant is being used for two different sessions you run the risk of archiving the same data twice, which could lead to erroneous results in the statistics of the archived data. If you have chosen a variant that has already been used, Archive Administration will issue a warning.

However, a warning of this kind merely alerts you to the fact that you may be archiving the same data more than once. You can still continue to archive the data. Make sure you think over how you proceed with your archiving. The program does not check whether the selection values overlap in variants with different names. It is possible that when you schedule your write job with these variants, that you are actually archiving the same data more than once, without receiving any special warning.

When you execute the write program ADK first generates an archiving session. Then the write program selects all the data that belongs to the first logical business object. As soon as the first business object has been written, ADK creates a new archive file. The next business objects are written to the next archive file. If the predetermined file size has been reached and there are still business objects to be written, ADK closes the current archive file and creates a new one. A business object is never divided and written to two different archive files.

This ensures that all data of a logical business object is physically saved together in one archive file. A business object can either be read completely or not at all. This prevents any inconsistencies from occurring later, when the data is deleted.

After the data of the last business object in your selection has been copied, ADK closes the last archive file. From the ADK point of view the writing phase is a purely technical process in which it does not matter whether the data originated from a single database table or from several different tables. This knowledge is stored in the archiving object on which a particular archiving session is based.

- The write process ends when one of the following events occurs.
- Archiving has been completed.
- The archive file has reached the maximum size that has been set in archiving-object-specific Customizing.

The maximum number of business objects in the archive file, set in archiving-object-specific Customizing, has been reached.

If, during the write phase, you find that there is not enough available storage space for all the archive files or if you know that you only have a certain amount of space available, you can interrupt the write phase and continue it at a later point in time. For more detailed information about this process see Chapter 3.2.3 "Interrupting and Continuing the Write Phase".

2.2.1.1 Archivability Check

To ensure that you do not archive any data that still belongs to currently open business processes, it is necessary to run an archivability check before the write phase. The check makes sure that the data to be archived meets the archivability criteria and can therefore be removed from the database. Which archivability criteria are used for a specific business object type mainly depends on the application in which it was created. Generally we can say that a business object, such as a sales or material document, is considered to be archivable if it

- has been completed,
- has reached the residence times entered for it, and
- is no longer used as a basis for other business objects.

The archivability check takes place either directly in the write program or in a special check program that has been scheduled as a preprocessing program. Which of these methods is used depends on the archiving object and cannot be determined by the user.

- Checks in the write program

  For a write program to be able to carry out the archivability check, it has to contain the entire technical and business process logic of a check program in addition to its own write logic. This allows it to only select those business objects that meet the archivability criteria and to only write these to the archive.
Check in the preprocessing program

In some cases the archivability check is carried out by a check program that runs before the write program and marks the archivable business objects with an indicator. Depending on the application this indicator is called the deletion flag or deletion indicator. Also used is the expression *To be archived* indicator. This type of check program is scheduled in Archive Administration as a preprocessing program for a specific archiving object.

With some SAP solutions, such as mySAP CRM, you have the additional option to run an archivability check for several archiving objects at once. This check is carried out by a check program that is independent of ADK and runs in the background. It marks the checked business objects with anarchivable indicator.

2.2.2 Deleting Data from the Database

After the business objects have been completely written to the archive during the write phase, they have to be deleted from the database in an additional step. Deletion is performed by an ADK delete program, which you can schedule using Archive Administration. To schedule the delete program you must select the archiving sessions that it is to process. Generally this only involves the most recently created archiving session. The delete program can be set to start automatically after a specific event has been triggered. For more detailed information about this process see Chapter 3.2.4 “Options for Automating Dependent Processes”.

If necessary you can change the sequence of the archiving phases, so that the data is first stored and then deleted. For more detailed information about this process see Chapter 2.2.3 “Storing Archive Files”.

At the beginning of the delete phase the delete program opens an archive file and then deletes the data that was stored in this file from the database. The deletion of the data is always done by object. However, before a business object can be deleted, the appropriate ADK function module verifies that the contents of the business object in the archive can be read. This procedure, called a *check read*, guarantees that business objects are deleted from the database only if they were properly stored in the archive file.

The check read and delete processes then continue sequentially until all the business objects in the archive file have been processed. For other files that belong to the archiving session you must schedule additional delete jobs, because with each execution of a delete program exactly one archive file is deleted.

Depending on the Customizing settings and their intended effects, there are two different scenarios for the execution of a delete program:

- **Deleting after writing**
  
  In this option the delete job is scheduled separately after the write phase is completed. The write phase is completed after all business objects have been written to archive files and the last archive file has been closed correctly. The deletion of the data from the database then takes place in a second, separate step. This scenario is particularly useful if you want to perform some other manual operation between the write and the delete phases, such as saving the archive files before deletion takes place; it could also be useful, for example, if your next available time window is not sufficient for the resource-intensive delete phase.

- **Deletion in parallel with writing**
  
  This scenario is useful if your main concern is performance. As soon as the write program has closed the first archive file (and while it is already writing the next one), the delete program check reads the first archive file and if the check is positive, it deletes the corresponding business objects in the database. The ADK runtime system automatically takes over the scheduling of the corresponding delete jobs. Because deleting data from the database generally takes longer than writing the individual archive files, several delete programs run simultaneously to process the archive files. As a result of this parallelization effect, the throughput of data is increased during the deletion process.

For detailed information about how to set up this type of scenario see the archiving documentation available in the SAP library (see appendix).
2.2.3 The Storing of Archive Files

The storing phase is an optional phase in the data archiving process. The actual data archiving phase is completed after the archive files have been written and the corresponding data has been deleted from the database. How the phases are supposed to be executed has to be decided by each data archiver, according to his or her company's requirements.

Generally it is not sufficient to write application data to the archive and then remove it from the database. The archive files themselves must be made available, so that the data they contain can be accessed. In general you need a storage system, and if you manually store your data, then you need a strategy for managing and securely storing the archive files. The storage phase can begin as soon as one or more archive files have been generated by the write program and closed correctly.

Choosing the right time to store your data

You can determine when the archive files are to be stored in archiving-object-specific Customizing in the Sequence area. Which option you choose is mainly determined by security issues. For example, if you choose **Store Before Deleting** the data is only deleted from the database once the archive file has been stored in a storage system. If you set the **Delete Program Reads from Storage System** indicator you can increase data security, although performance may decline. In this case the check read of the data during the delete phase takes place directly from the storage system and not the file system. In other words, before you delete you can double check that the archive file was stored successfully.

To store archive files you have the following options (for more detailed information on this topic see Chapter 4 “Storing Archived Data”).

- **Storage system**
  If you are using a third party storage system connected via the ArchiveLink interface (see Chapter 4), then you can instruct it to store the archive file at the end of the write or delete phase. For this to occur ADK sends a command to ArchiveLink, which controls the communication with the storage system. The storage of files takes place either when the write phase is completed or not until the delete phase was also completed. You can set the sequence of events in archiving-object-specific Customizing. The storage phase can also be triggered manually.

  When you do, ADK sends a command to ArchiveLink, which controls the communication with the storage system. The storage of files takes place either when the write phase is completed or not until the delete phase has also been completed.

- **HSM systems**
  With this type of storage the archive files are created directly in the file system which is linked to the storage hierarchy of an HSM system (see Chapter 4). The HSM system independently takes care of the storing and management of the data. Communication does not have to take place via ArchiveLink. Neither does the storage process have to be controlled from the SAP system side. All you need to do is enter the path to the target file system in Basis Customizing under **Cross Client File Names/Paths**. The HSM system is set up in such a way that the user can access the data there as if it were located on a local system.

  From the point of view of the user the data storage using an HSM system is transparent, meaning that he or she cannot see where the data is located. The only indicator that the data being accessed may be archived and therefore located on a slower medium, would be a slightly slower response time.

- **Alternative storage media**
  If you prefer not to store your data on a storage or HSM system, you can also manually move it to tertiary storage media (magnetic tapes, CD ROMs, optical disc, etc.) or store it using standard backup mechanisms (backup, mirroring, etc.). The stored archive files are then managed by your IT department.

  This type of storage may be less costly and easier to implement, but it requires more maintenance and management efforts. In addition you would need to implement a comprehensive management strategy to ensure the safety of the data (for example, periodically moving the data to a new, more robust storage media).

  To be able to manually store your archive files on tertiary storage media, the archive files must have been closed correctly during the write phase and the automatic storage of the data has to be switched off.
However, you cannot manually store your data before or during the delete phase, because the delete program must be able to access the archived data.

A great disadvantage of the manual storage of archived data is the fact that the management of this data is usually also performed manually. This makes it considerably more difficult to automatically find and retrieve the archive files later.

2.2.4 Other Processes and Functions

In addition to the write, delete and storage procedures we have mentioned, and which are considered to be the core functions of data archiving, there are other secondary processes and functions that are used less frequently and do not apply to all archiving objects.

2.2.4.1 Displaying Archived Data

Data archived with SAP data archiving is stored in such a way that it can always be displayed later, with the help of the appropriate read program. The read programs are provided by each archiving object and are used to select the archived data based on the selection criteria entered by the user, and to display them in an appropriate format.

Access to archived data occurs with the help of meta data (see below), which ADK writes to the archive file together with the application data during the write phase. When the archive file is accessed this meta information is interpreted and used to display the data. During this process the data in the archive files is not changed in any way.

There are several ways you can access archived data. In principle it is possible to read every archive file that has been created in the same system and client from which it is being read. How technically sophisticated a read program of a specific archiving object is depends on the programs provided by the applications.

The following options exist for displaying archived data:

- **Evaluating several business objects**
  With this access method, also called sequential reading, all the business objects in an archive file are read sequentially and are displayed in a list format, according to the selection criteria entered by the user. The program generally only displays the most important information of the business objects, such as customer, order date, or item number. This is the simplest way to display archived data and is mainly used for evaluation purposes. This type of evaluation can, for example, span across all documents of a specific posting period or a specific document number interval. The amount of time it takes for the read program to read the archived data depends on the size and number of archive files to be read. Most archiving objects support this access method.

  In addition, some applications offer the option of a **combined evaluation**, meaning that it is possible to display data that resides in the database together with data that has been archived. Examples of such programs are RFBEI00 for creating the compact journal and RFEP0J00 for creating the line item journal. Both are part of the SAP R/3 application Financial Accounting (FI). They list the most important header and items data for the selected financial accounting documents, such as the account, document or business volume type, or the booking period.

  For both evaluation programs you can choose the data source, which determines whether the program reads the documents only from the database or also from the archive.

- **Display of individual business objects**
  This type of access is also called direct access, single document access, or single document display. You can use it to search for and display complete business objects, from, for example, sales documents or financial accounting documents. The single document display requires an index, which tells the read program the exact place where each business object is stored in the archive. The storage location is defined by the key of the archive file in which the business object is located, and the exact position of the object within the archive file, which is called the offset. With this information the read program can find and display the requested business object.

  You have the following options to build the index needed to display the archived data: For some archiving objects, such as FI_DOCUMNT used to archive financial accounting documents, you can build or delete the index directly from Archive Administration using the Index function. The index can be built before and after archiving.
This, however, is a relatively simple index and is not included in all archiving objects. The selection options for searching for archived data are rather limited with this type of index. A much more comprehensive and comfortable measure for searching for and displaying archived data is the Archive Information System, which you can also reach from Archive Administration.

For more information about the different options and tools available for accessing archived data see Chapter 5.

2.2.4.2 Scheduling pre- and postprocessing programs

Some archiving objects offer pre- and postprocessing programs in addition to the write and delete programs that all archiving objects must offer. The purpose of these additional programs can vary from one archiving object to another. The following section describes the purpose of pre- and postprocessing programs and how they are used in different real life scenarios. If an archiving object offers one or both of these programs you can operate them using the functions Preprocessing or Postprocessing.

- **Preprocessing program**
  A preprocessing program is mainly used to check the selected business objects for archivability and to set an archivability indicator or a status for each checked business object (see above). Business objects which have received an archivable indicator can no longer be changed. If the user calls up this business object in the change mode, he or she receives a message window telling him or her that the business object has been marked for archiving and can therefore no longer be changed. The function of the write program later is to simply select the marked business objects and to write them to the archive without performing any other checks. This concept of splitting tasks between the check and the write programs is advantageous from a performance perspective and allows the archiving processes to be better integrated in overall systems operations. It is mainly used for data archiving in mySAP CRM.

  Another example of how a preprocessing program is being used is the SAP R/3 application Sales and Distribution (SD). In the archiving process of sales documents the preprocessing program is used as an analysis program, which determines the number of archivable documents, but does not mark them with an archivable indicator.

- **Postprocessing program**
  Not all archiving objects offer postprocessing programs. They are used to perform certain tasks in the database that may be necessary after an archiving session has been completed. Such tasks may include the removal of log entries, cleaning up index tables and updating statistics data. Here the postprocessing program serves as a "clean up" program for the remaining data after archiving. Postprocessing programs operate only on database data, not on archived data. Therefore it is generally not necessary to be connected to the archive. Postprocessing programs are generally executed after an archiving session, but they can also be used separately from data archiving.

  For example, after you have archived financial documents with archiving object FI_DOCUMNT, it may be useful to delete the secondary indexes for which the runtime has been exceeded. This is done by an index delete program which is automatically started in the background after the last delete program has finished. To start the program you must first make the appropriate settings in archiving-object-specific Customizing. If these settings for the automatic start of the program were not made, you can start it manually, if you are facing an acute lack of memory space in the database.

  In some applications post processing programs are used to delete previously archived business objects from the database. In the SAP R/3 application Quality Management (QM) for example, the data of the archived inspection lots is removed from the database with the help of a postprocessing program.

2.2.4.3 Reloading archived data

Even after many years and upgrades to new releases it is possible to display data archived with SAP Data Archiving, using the appropriate read programs. Therefore, in most cases it is not necessary to reload data back into a production database, and should only be done in emergencies.

An example of an emergency would be a situation where you realize after you have archived that you have archived data that is still needed in the database. Such an error may occur due to wrong Customizing settings, choosing a residence time that is too short, or entering the wrong selection criteria. If you detect an error you should reload the data that was accidentally archived back into the database immediately after archiving. A reload of this kind is generally unproblematic. However, with some archiving objects it may be that not all data that was archived can be reconstructed in the database. When you reload a sales document,
for example, the reload program cannot reload the cost center debits (controlling data) that are linked to the sales document. This means that the data will not exist in the database.

Also, there are certain tables in data archiving from which data is only deleted. This means that the data deleted from these tables during the data archiving process, cannot be reloaded. Another restriction involves data from certain archiving classes, which cannot always be archived. Before you reload archived data, you must seriously consider the consequences that this option may bring with it.

This is especially important if you are considering reloading data that has been archived for some time already. In such a scenario the risk that the archived data and the data in the database no longer coincides is very high. This has to do with the fact that the data would be reloaded from a historical context into a current database context. You may be overwriting documents that belong to a document number interval that was reset between the time of archiving and the time of reloading. Reloading can also affect the consistency of organizational data, because you may be reinserting data into the system that no longer exists.

Thus, the longer the amount of time between archiving and reloading, the higher the risk of encountering data inconsistencies after the reload. Reloading across releases as a rule should never be done. If you do, SAP cannot be responsible for any problems that may occur.

2.3 Important Data Archiving Features

2.3.1 Data Security

To ensure that no data is lost due to errors during the archiving session, the data archiving process consists of two steps. In the first step the data is written to the archive files. In the second step the data is removed from the database. However, this only takes place after the archive file has been written in its entirety and read successfully. This process helps detect any errors that may have occurred when the data was transferred from the database to the archive file via the network. If an error occurred you can set up a new archiving session, because the data is either still located in the database or in an archive file.

2.3.2 Archiving in Online Mode

An archiving session, made up of write and delete jobs, can take place while the system is online, meaning that users can continue to use the system while data is being archived. However, you may encounter performance bottlenecks if tables from which data is being deleted are also accessed in online business processes. Therefore it is recommended that you archive during times of low user activity in your system.

2.3.3 Data Compression

During data archiving data is compressed by up to a factor of 5. Data that is saved in cluster tables is not compressed any further. Through the compression the archive files occupy as little space as possible on the storage system.

2.3.4 Release- and Platform-Independent

To ensure that the archived data can be interpreted over long periods of time, ADK also stores metadata in the archive file together with the application data. This meta data contains information about the current runtime environment of the data, such as:

- The schema of a database table
- The data type and length of a table column
- The codepage (ASCII, EBCDIC)
- The number format (such as integer on various hardware platforms)

ADK uses the metadata to take into account changes to the database structures (field types, field lengths, new fields) that may have occurred over time. This adaptation is only temporary, in that it is only in effect during the read access to the archive file. It does not mean that the data in the archive file is changed in any way. With this approach it is not necessary to convert archive files because of hardware or software changes. Thus, SAP data archiving is able to ensure the release- and platform-independence of that data.
3 Technology and Administration

3.1 The Basis Technology of the SAP Archiving Solutions: ADK

3.1.1 ADK Positioning and Components

SAP data archiving can be used with all database systems that are supported by SAP solutions. This database-independence is achieved through the archiving technology, which ensure that the flow of data takes place “above” the database interface. This approach is called database-based archiving. With this approach it is possible to use different database systems and still ensure that the business links between the data to be archived are maintained.

Although data archiving programs may vary from application to application in terms of how they select the data to be archived and check logic, all archiving programs use the same basis services. These include services to compress the data contained in the archive files and services used to read the archived data independent of which release is currently being used, among others. The software layer that provides these basis services used to develop and run archiving programs is the Archive Development Kit (ADK), which is part of SAP NetWeaver.

ADK is not only comprised of the ADK runtime system, but also offers an administration environment (transaction SARA and DB15) and an environment for the definition of archiving objects and archiving classes (transactions AOBJ or ACLA). Management and meta data is managed in the ADK Repository, which consists of ADK tables of the same SAP database.

3.1.2 ADK Runtime Environment

Figure 9 shows the interaction of the database, the archiving programs, the ADK components and the file storage as a function of the data and control flow during the write phase:

Figure 9: ADK as a Runtime Environment for Archiving Programs

The write program is scheduled in the form of a write job in archive administration. Within the write program an ADK call generates a new archiving session that is entered in the ADK repository. The application data, which is read for a specific archiving object and checked for archivability is transferred record-by-record to ADK and bundled into data objects by the ADK functions.
Data object services

Other ADK internal data object services transform and compress a complete data object into a format that is platform-neutral and can be read across releases. Depending on the data, it is possible to achieve a compression rate of 1:10, or even higher, if the data records contain many initial values.

Before the data object can be written to a file, ADK makes sure that the meta data necessary for the subsequent technical interpretation of the archive files has been transferred from the ADK repository and ABAP dictionary. This particularly includes the Name Tabs of all tables and structures that belong to the archiving object.

When it accesses the archive the ADK runtime system checks whether the following conversions are necessary due to a changed system environment. If yes, it carries out the corresponding conversions.

- **Platform adjustment**
  If the codepage or number format have changed, a platform adjustment has to be made. When archive files that originated in a non-Unicode system are being read in a Unicode system (see background information in the information box below), a codepage conversion always takes place for the character-like data.

- **Schema adjustment**
  If the archived tables and structures have been changed with respect to their original definition in the ABAP dictionary, a schema adjustment has to be made. The structural changes, however, must ensure compatibility of the structure components. This is accomplished by using the same semantics as in the ABAP command MOVE-CORRESPONDING. This means that structural components that did not exist when the data was archived are returned with initial values, and components that are no longer used after the upgrade are not output. If components have the same name then the usual ABAP conversion rules between the different data types apply.

---

**Data Archiving and Unicode**

Unicode is a universal character set used to facilitate a better exchange of data between different systems and across boundaries. It helps prevent problems that may arise when communicating systems use different code pages. These problems mainly arise, because a code page is only applicable to specific languages and cannot be combined at will with other code pages. Using Unicode helps you avoid these problems.

What are the conditions necessary for data archiving to work in a Unicode system? Generally it is sufficient if ADK and the archiving objects and archiving classes used adhere to the stricter ABAP syntax. ADK as part of the SAP Web Application Server meets these Unicode requirements, as do the SAP archiving solutions described in this document. However, any modified or customer-specific programs must be checked for their Unicode capabilities, and if necessary, must be converted. For help on this topic read the chapter "ABAP and Unicode" in the SAP Library, the ADK documentation (especially the function module documentation, which mentions the new parameter RECORD_REF used for reading data records), and the example programs delivered by SAP, SBOOKR, SBOOKR_2 and SBOOKR_3.

If you want to convert your SAP system to Unicode, you can speed up your conversion by first removing data from your system through data archiving.

Archive files that were created before the Unicode conversion are treated as a special case of the temporary platform adaptation (see above) that is performed automatically. Therefore, you never have to convert any archive files, because of a Unicode conversion.

For more detailed information about Unicode see SAP Note 73606 and 379940, or the website [http://www.unicode.org](http://www.unicode.org).

These conversions are carried out only temporarily during the runtime of the read, delete and reload programs. Archive files are never actually changed. If, however, the changes in the system environment have been more comprehensive than described above, it is possible for you to use ADK to implement special conversion programs, to permanently convert your archive files.

**ADK File Management**
With the other runtime system services listed in Figure 9, ADK takes over other technical and application-independent tasks, so that the archiving programs are not responsible for these tasks. ADK file management makes sure that new archive files are automatically created and named during the write phase, as soon as the limits set in archiving-object-specific Customizing have been reached. Choosing a syntactically correct path used later to access the archive file is not part of the logic of an archiving program.

3.1.3 ADK as a Development Environment

ADK functions as a runtime environment for all archiving programs, but it also serves as a development environment. The main development component of ADK is the application programming interface (API), which comprises a series of function modules of the function group ARCH. ADK provides this API for all archiving objects.

ADK API is used for the following tasks:

- **Developing new archiving objects**
  
  Archiving solutions for SAP standard business objects are developed by SAP. Developing a customer-specific archiving solution within the customer namespace basically means using the ADK API function modules to define an archiving object and create the necessary archiving programs.

  The standard SAP archiving solutions are also developed with the ADK API.

- **Expansion of archiving objects**
  
  It may be necessary to expand and SAP standard archiving object, if, for example, you created customer-specific append structures for standard tables, or you have your own tables that logically belong to SAP standard tables. Depending on the type of change to the data structure you want to make, expanding a standard archiving solution may require some more comprehensive modifications to the delivered standard.

  Before you make any changes to standard objects, make sure you have carefully analyzed whether or not there are any dependencies to other standard archiving objects. You must also check whether or not, from a business processes point of view, it will be necessary to run additional archivability checks for the affected objects. If you have doubts, contact an experienced consultant or SAP consulting.

The features of ADK are described in detail in the SAP Library and in the documentation for the individual function modules. Another good source of information about how to develop customer-specific archiving solutions are the SAP Training Course BT670 and the ADK White Paper (available in SAP Service Marketplace, /ADK).

3.2 Tasks of the Data Archiving Administrator

3.2.1 The role “Data Archiving Administrator”

ADK offers a pre-defined role, which gives a system administrator, for example, easy and personalized access to all important information, tools and systems, he or she needs for his or her daily work.

The role “Data Archiving Administrator” is available as of SAP R/3 4.6 C. It encompasses a much wider range of tasks than those involved with the central transaction SARA. Figure 10 shows the menu of the role, how it appears in the role maintenance transaction PFCG, and how it can be adjusted. The technical role name is SAP_BC_CCM_DATA_ARCHIVING.
This role contains the transactions you need for the following tasks and links to corresponding websites in the SAP Service Marketplace:

- Selection of the appropriate archiving objects based on an analysis of the database and table growth (DB15, DB02, SE16).
- Planning, controlling, monitoring and evaluating archiving sessions (SARA, SAR_SHOW_MONITOR, SARI).
- System settings, especially the Customizing of the storage of archive files (SAR_OBJ_IND_CUS, FILE, SF01).

In this document we are not able to describe each task of the data archiving administrator in detail, due to space and time limitations. We therefore want to focus only on a few important tasks in the following section. For more detailed information about all tasks that are part of data archiving, see the SAP Library.

3.2.2 Analysis

Large data volumes present a greater burden on the capacity of the database and backups. In addition, in many high availability scenarios data is often replicated and mirrored. This also means more investment in the necessary hardware for these scenarios. The costs for backups go up even more dramatically as data volumes rise - the amount of time needed for a backup goes up proportionally to the increase in the data volume.

The analysis phase is the most important part of an archiving project. An analysis can be approached from two different viewpoints:

- From a technical viewpoint
- From a business viewpoint

3.2.2.1 Analysis from a Technical Viewpoint

This type of analysis focuses on the database size and the growth of the database tables. This technical analysis must then be compared to the business view analysis.

In addition, you must identify the archiving objects that are linked to the critical database tables. Before this step you should check whether there is anything you can do to reduce the data volume that is to be archived.
through data management measures (see Chapter 1.4). The individual user departments know best which data is no longer needed and therefore do not need to be archived.

It is difficult to make general statements about when is a good time to remove data from the database. It is part of the system administrator's tasks to determine specific criteria that can be used to decide whether data can be removed from the database or not. Also part of the routine tasks of the administrator are the monitoring of the database and ensuring that the production system's performance remains intact. For this purpose SAP has provided the administrator a series of powerful tools. We assume that if you are a system administrator that you are familiar with different monitoring tools and how to interpret the values they show. The following section we give a short introduction of each tool. The corresponding transactions or programs are listed in the parentheses.

- **Database Monitor (DB02)**
  You can use the database monitor (transaction DB02) to determine the size of tables and how these tables have grown in the past. This transaction returns important database-specific figures. The display of the database monitor depends on the database you are using. With an Oracle database system, for example, the monitor displays the number of free table spaces or the size and the growth of the individual tables and indexes.

  In addition to transaction DB02 you can also use the tool SAPDBA and transaction ST03 (performance monitor), to find out other indicators that could help you with your data archiving decisions. You can also view memory space statistics using transaction DB15, by choosing *Space Statistics*.

- **Table Analysis (TAANA)**
  You can use this function to run analyses for database tables and analyze the corresponding table entries. This analysis function determines how many entries in a given table exist for a specific field value combination. It does this with the help of analysis variants, which contain the corresponding field lists. The analyses help you determine which archiving objects you need to use and can indicate which selection criteria would be the most effective during archiving. It also helps you avoid having to analyze archiving objects or organizational units that do not generate large amounts of data. For more detailed information see the SAP Library.

- **Application Analysis (ST04)**
  You can use transaction ST04 to carry out application-specific analyses of table contents. You can choose the application you want to analyze on the entry screen. Then you can schedule different analyses and then analyze them. These analyses can give you important information about document size and document type runtimes.

- **Tables and Archiving Objects (DB15)**
  After you have identified the critical tables, you must find out which archiving objects these tables belong to. You can see this with the help of the function *Tables and Archiving Objects* (transaction DB15).

  This function shows you which tables belong to which archiving objects and vice versa. This will enable you to assign the tables you identified in the technical analysis to specific archiving objects, which can then be used to archive the table entries. This transaction provides the following functions:

  - Display of all the tables belonging to an archiving object and from which entries are to be archived
  - Display of all the tables belonging to an archiving object and from which entries are deleted
  - Display of database-specific space statistics, such as number of records in a table, how much space a table occupies in the database (in KB) and other table details from the SAP and/or database statistics.

**3.2.2.2 Analysis from a Business Viewpoint**

With this type of an analysis the table entries are analyzed from a business point of view using standard analysis tools and then they are assigned to the archiving objects. In principle business objects can be archived independently of each other. However, there may be dependencies that require that the data is archived in a specific sequence. For more information on archiving sequence see the Network Graphic (in Archive Administration under *EnvironmentÆNetwork Graphic*).

You may have to analyze your business processes to determine which processes generate large data volumes. Here it is important that you consider the individual objects in the context of the actual component
running in the production system and the corresponding business processes. The following section describes the individual steps that make up this type of an analysis.

*Identify the data that can be archived*

From a business point of view it is mainly important to determine which data can be archived when. You can only archive those business objects that belong to closed business processes. To be able to do this you must first determine when a business process is considered to be closed. This depends on the application to which it belongs. Here it is also important to take into account the requirements of your company, such as different runtimes for a business object based on its organizational unit (company code, plant, sales organization, etc.). Besides the analysis of the business processes the CO analysis programs RARCCOA1 and RARCCOA2 (see the program documentation for more detail) are among the most important tools used to identify the archivable data.

In addition, you should determine whether or not data exists that is no longer needed from a business point of view. This data can be for example:

- Legacy data from data migrations
- Data from separated business areas (because the area was sold, for example)
- Test data

Keep in mind that this data can often belong to business processes that are not yet closed. Therefore you should plan enough time for the data removal in these cases.

### 3.2.3 Monitoring

An important task of the data archiving administrator is monitoring the archiving sessions. You can use the Archive Management function for this task and as of SAP R/3 4.6 also the Data Archiving Monitor. The archive management function is available in Archive Administration (transaction SARA).

**Archive Management**

This function gives you an overview of all archiving sessions and archive files for a specific archiving object. You can use it to display current information about the archiving sessions, the archive files, and the archiving jobs. In addition from the function you can branch directly into the spool list of the write job, to Data Archiving Customizing, the archiving objects and tables (transaction DB15), the Archive Information System, and the data archiving statistics.

The job overview displays the archiving sessions for a specific archiving object according to their status. Below we have listed some of the most common statuses for archiving sessions. For a description of the additional statuses see the SAP Library, or the Legend in archive management.

- **Incorrect Archiving Sessions**
  The write process was terminated before the first archive file was completed.

- **Incomplete Archiving Sessions**
  The write process has not yet completed, the delete program has not yet run for all archive files, or a write process was terminated.

- **Complete Archiving Sessions**
  Both write and delete phases are completed.

- **Archiving Sessions to be Archived**
  The management data of the archiving session can be archived and deleted with archiving object BC_ARCHIVE.

The archiving sessions are listed according to their status. Within a status area the sessions are grouped into blocks of 20. When you expand an archiving session you can see a list of the archive files that belong to the archiving session.

- **Archiving Session Details**
  You can double click every archiving session and each archive file to display the details for each. For the archiving session the date, time and user are displayed, among other information. In the change mode of
the detail screen you can enter an archiving session note and set the To be Archived or the invalid indicators. The management data of the archiving sessions that have the To be Archived indicator activated, is archived the next time the archiving object BC_ARCHIVE is used. You then no longer have read access to the archiving sessions whose management data was archived.

• Archive File Details
The detail screen for the archive file also contains information about the size of the archive file and the number of objects it contains. If the archive file has been stored in a storage system, the field Storage System will contain the status Stored. In the change mode you can change the name of the archive file and the logical path. You can also enter a note and a long text for the archive file.

If a name appears for the archive file, the system assumes that the archive file is in the file system. If this is the case the system will check the accessibility of the archive file in the file system, when you call up the detail screen. The result of this accessibility check is displayed in the last line of the detail screen. If the check was positive, the status Archive File Is Accessible (represented by a green traffic light) is displayed. If not you will see the status Archive File Is Not Accessible (red light). If the archive file has been stored in a storage system and no file name has been entered for the archive file, the system will run a check to see whether or not the file in the file system can be accessed. The file then receives either the status Archive File Is Accessible in Storage System (yellow light) or Archive File Is not Accessible (red light).

Archiving sessions or archive files with terminated archiving sessions are represented by a lightening rod icon. Archiving sessions that have not been completed are represented by a clock icon. In the detail screen of the archiving session or the archive file you can see the names of the jobs that belong to the file or the session. You can double click the archiving session to display an overview of the job. Here you can monitor the status of the job. In this screen you can also see the job log, the spool list, and details for the job.

Choose the function User Entries to branch to the selection criteria that was chosen for this particular archiving session when the write job was scheduled. These selection criteria will be displayed for this job, even if the variant no longer exists.

Choose Goto → Stored Files to display an overview of the archive files that have been written to the file system. You can restrict your selection by for example choosing stored archive files for all archiving objects or for a specific archiving object or archiving session. The results list shows the content repository in which the file is located and the technical key of the archive file in the storage system. In addition, the overview contains information about the status of the archive file in the storage system.

Data Archiving Monitor
The CCMS Alert Monitor is a central monitoring tool for the entire IT environment (particularly for SAP systems). It comprises a series of dedicated monitors for individual system components. A system administrator who has already monitored system availability and throughput using the Availability and Performance monitor, or has used the Background Processing, File Systems, or Knowledge Provider monitors in the context of data archiving, will know how valuable it is that data archiving has been integrated into the monitoring infrastructure. The monitoring infrastructure is even more useful when auto-reaction methods have been implemented. They can send an e-mail or an SMS message, for example, in case there is a problem.

The data archiving monitor offers the following archiving-specific functions for monitoring the processes, recognize problem areas and analyze problems:

• A general overview of all archiving sessions (here you do not enter an archiving object first, as you would in archive management).

• Progress reports about the processing of the archive files.

• Compact information about the technical details of the write and delete jobs, such as starting time, runtime, size of the archive files, and number of archived data objects.

• Alerts to indicate a possible need for action (for example, yellow alerts for delete jobs that have not run yet or incomplete delete jobs, and red alerts for certain error situations).

• Provides a link to the jobs that triggered the alert and their job logs, to help you analyze the warning.

The data archiving monitor is not able to report all potential runtime exceptions (such as a job termination due to a system error that took place outside of ADK). We therefore recommend that you use the data
archiving monitor together with the other monitors we mentioned earlier, or that you configure a monitor that is specifically configured for your requirements.

You can call up the data archiving monitor from the CCMS Monitor Collection (transaction RZ20) from the menu SAP CCMS Monitor Templates → Data Archiving, directly by using transaction SAR_SHOW_MONITOR or from the role we described earlier. For documentation go to Application Help or use the long texts that describe each node. You can also use the information and hints provided within the monitor itself.

3.2.4 Settings

3.2.4.1 Overview of Customizing

The settings for the archiving processes are made in archiving Customizing, which you can reach from Archive Administration. The following section will give you a general overview of what you can do in data archiving Customizing. For more detailed information and examples see the corresponding documentation in the SAP Library.

Data archiving Customizing is divided into the following areas:

- **Cross-Archiving-Object Customizing**
  The settings you make here affect all archiving objects across the board.
  - **Data archiving monitor**: This tool provides the data archiving administrator with archiving-specific functions to monitor processes, and recognize and analyze problems. This includes for example the progress indicator for processed archive files, a general overview of all archiving sessions that have been processed for all archiving objects, and alerts that point you towards current or potential problems during archiving. See also “Data Archiving Monitor”.
  - **Access check during archive selection**: Archive files for which the access check returned negative results, are represented in the selection screen with a lightening rod symbol.
  - **Verification of archive files**: During the write phase the program writes verification information into the archive file for every data object. With this information an archive file can then be “check read” before the actual procedure, such as deletion, reading, or reloading, actually takes place, to make sure that it and all of the data objects it contains has remained intact.
  - **Automatic interruption of the write phase**: It may be necessary to interrupt the write phase due to low system resources, or because the time frame for the archiving session is too short. The interrupted write phase can be continued at a later point in time. The interruption function is part of ADK.
  - **Server groups for background processing**: When you schedule archiving sessions in the background, you can enter the server groups on which the sessions are to be executed.

- **Archiving-Object-Specific Customizing**
  The settings you make here only apply to the archiving object that you entered previously in archive administration. Some settings, such as those for the post processing program, are only offered for those archiving objects that include such a function.
  - **Logical File Name**: Here you choose the logical file name under which the archive files are to be stored in the file system. At runtime a complete, platform-specific file name including the path is determined for the logical file name. The logical file name is maintained with transaction FILE. See “Basis Customizing”.
  - **Archive File Size**: Here you can enter the maximum size an archive file can reach before it is closed. The file size can be determined either by entering a size, such as 100 MB, or by specifying the maximum number of data objects an archive file is to contain. As soon as one of these values has been reached, the archive file is closed and a new one is created.
  - **Settings for Delete Program**: Here you can determine whether the delete program is to be executed in the test or production mode, and whether or not the delete program is to be started automatically.
- **Place File in Storage System**: If you are using an external storage system, you can enter the content repository here in which the archive files are to be stored. You can also determine here whether the archive files should be stored automatically and whether the storage step should take place before or after the delete phase.

- **Basis Customizing**
  Here you enter the file name and file paths used for the archive files. To define a logical file name you need a name and a path. The path again, is of a logical nature, meaning that there is a rule that determines at runtime (platform-dependent), what the physical path is to be, based on the information you entered. File names and paths can be either client-specific (transaction FILE) or cross-client (transaction SF01).

- **Application-Specific Customizing**
  With some applications it is possible to determine criteria for the archivability and deletability of application data. These criteria are then considered during archiving. Examples are account type life and document type life for financial accounting documents. These parameters are set in the Customizing tables of the application in question.

3.2.4.2 **Security Versus Performance**

Data archiving has to do with the processing of mass data and in this process it must be ensured that access to archived data is possible in a secure and efficient manner. It is the task of the data archiving administrator to ensure that the relationship between security and performance during archiving procedures is always optimally in tune with the requirements of the user departments that run the business processes.

**Verification of Archive Files**

An important performance issue is the amount of runtime needed for checking whether the archive files are intact. During the write phase the system stores verification information based on a CRC-32 (Cyclic Redundancy Check) checksum in the archive file. With this information an archive file can be "check read" before the procedure, such as deletion, reading, or reloading, actually takes place, to make sure that it and all of the data objects it contains has remained intact. This procedure recognizes and reports the incorrect archive files, and subsequent activities, such as the deletion of archived data from the database, are not started as a result. If you encounter incorrect files contact an SAP consultant or an experienced data archiving consultant.

You can determine when the verification is to take place in Cross-Archiving-Object Customizing. Keep in mind, however, that although you will achieve increased security by checking your archive files, you will also generate longer runtimes. Depending on the archiving object, they can be up to 10-30% longer.

**File Accessibility Check During Archive Selection**

During the selection of archive files for delete, read or reload procedures, you can have the system check that an archive file exists. This means that the system checks whether or not the archive file can be accessed by archive administration and whether the meta data of the archive file is readable. Archive files for which the access check returned negative results are represented in the selection screen with a lightening rod symbol.

What kind of access check is to be carried out is determined in Cross-Archiving-Object Customizing. You can let the system run checks on archive files that are still in the file system, and on archive files that have been stored on a storage system. If you do not need the access check at the time of the file selection, you should deactivate this setting in Customizing.

Running an access check during archive file selection can be very time intensive for archive files, especially if you have many files in the storage system. Therefore, you should use these checks only after careful consideration. The same thing is valid for archive files that are located in a file system, which is connected to an HSM system.

**Reversing the Order: Storing Before Deleting**

After the archive file is created during the write phase there are two possibilities for how the delete and storage phases should be executed. You can indicate which option you would like to execute in Cross-Archiving-Object Customizing:

- **Delete Before Storing**
The write phase is followed by the delete phase, during which the data is deleted from the database, based on the data in the archive files. The archive files are stored in a storage system in the following storage phase.

With this option it is important to ensure either that the archive files are written to mirrored disks or to a RAID file system, or that they are saved before the delete phase.

- **Store Before Deleting**
  The write phase is followed by the storage phase, during which the archive files are written to a storage system. The data is not deleted from the database in the subsequent delete phase, until it has been stored successfully.

In addition to these two settings, in Archiving-Object-Specific Customizing you can also enter a storage system (*Content Repository*) and indicate that the storage of archive files should be started automatically (*Start Automatically*).

Moreover, when you choose *Store before Deleting* you can also decide how the delete program is to read the data, with the indicator *Delete Program Reads from Storage System*:

- If you activate this option, the archive file is deleted after it has been stored in the file system. During the delete phase the delete program reads the data form the archive file in the storage system. This ensures that the delete program receives the same data as was transferred to the storage system.

- If you deactivate this option, the archive file is not immediately deleted after it has been stored in the file system. During the delete phase the delete program reads the data from the archive file in the file system. This helps improve the performance of the delete program, without sacrificing the early storage of the data.

To start the delete program automatically after the archive files have been created or after they have been stored, the setting *Start Automatically* has to be activated under *Settings for the Delete Program*. If you have chosen *Store Before Deletion* (and only in this case) it does not matter whether the delete program is started in the test or production mode. If you have chosen *Delete Before Storing* and you have activated the test mode variant, then no data will actually be stored.

The data archiving administrator must decide on a case-by-case basis in which order the delete and store phases should be executed. If security is the most important focus, then the delete phase should take place after the store phase. This ensures that the data will not be removed from the database until the archive files have been stored correctly. The process becomes even safer if the delete program is set to read the archive files from the storage and not the file system. If performance is your main focus, then the delete phase should occur before the storage phase.

### 3.2.5 Data Archiving Statistics

#### 3.2.5.1 Collection of Statistics

During the write, delete and reload phases ADK collects statistics, aggregates them and stores them in the database for later use. The data archiving administrator can use this information to

- Recognize resource bottlenecks in the system early on
- Better plan future archiving projects
- Document how data archiving has reduced the amount of data in the database, to show the benefits of data archiving

The statistics include the runtime of archiving jobs and the number of processed data objects, as well as information about the amount of disk space used by the archive and the amount of space that was cleared in the database.

The amount of space taken up by the archive files can be calculated relatively exactly. The amount of space cleared up in the database is calculated based on data from the ABAP dictionary and is therefore more a general indicator for how much space was actually freed from the database. If you want to use the database space that was freed up through archiving you must reorganize your database tables. For more information about database reorganization after data archiving see Chapter 3.1.7.

#### 3.2.5.2 Displaying Statistics
The data archiving administrator can use the following tools to display statistics:

- **Analysis Transaction**
  You can use the analysis transaction to display all statistics that were collected during data archiving for all archiving sessions. You can call the transaction either directly from the initial screen of Archive Administration or from Archive Management. You can also use transaction SAR_DA_STAT_ANALYSIS.
  
  To select an archiving session to display you can use the client, archiving object, archiving date and also the status of the archiving session. To further process the statistics, the data archiving administrator can use the comprehensive functions of the ALV Grid Control (SAP List Viewer), such as print or export.

- **Standard Log**
  The standard log outputs the statistics collected during write, delete, read, and reload sessions in the form of a screen list if the programs were run in the dialog mode and a spool list if they were executed in the background. For the write program the standard log can be output in the production and in the test mode; for the delete, read and reload programs the standard log can only be output in the production mode.
  
  In addition to the already mentioned statistics, the standard log also includes detailed information about the number of processed structures.

### 3.2.5.3 Archiving Statistics Data

The statistics that are collected during archiving are saved in database tables and can be archived together with ADK management data, using the archiving object BC_ARCHIVE.

### 3.2.6 Reorganization of the Database After Data Archiving

Data archiving is mainly used for two reasons:

- On the one hand it is to improve the state of the database system, such as response time, I/O rates and the quality of the data buffer.
- On the other hand, it is used to free up occupied space in the database.

The space freed up in the database during the delete phase can usually not be used immediately for new data. Only those data blocks that have been emptied below a certain limit can be used for new data. If you want to use the space freed up in other data blocks that are not below this limit, you must reorganize your database tables after data archiving.

In the following we will explain how data archiving affects the database system and what you need to do to achieve the above objectives. Because different systems use different architectures and terminology, we will focus on the example of Oracle databases in this section. However, the measures we explain can be applied to any database system. As mentioned, during the delete phase large amounts of data records are removed from database tables, which leaves both the database blocks and the tablespaces highly fragmented. This fragmentation is represented in Figure 11.
Figure 11: Database Blocks and Tablespace Fragmentation

To help you avoid the negative effects of fragmentation on your database system, you can choose from index, table and tablespace reorganization.

3.2.6.1 Index Reorganization

Of all the reorganization options index reorganization is the most important measure to help you improve the performance of your database. To keep the time needed for database accesses short, you need a higher buffer hit rate for the data you are trying to read. Especially with indexes the probability that the data is already in the data buffer should be close to 100%. This requires that the indexes use up as little working memory as possible.

Figure 12: Index Fragmentation

When data is archived or even simply deleted, gaps are created in the indexes of the affected tables; see Figure 12. The database system may not be able to fill these spaces again, because the order of entries in
the index predetermines which data blocks can be filled. If there is no free space in these data blocks, new data blocks have to be used. As a result and despite data archiving, the need for more disc space for these indexes continues to grow. This in turn will have a negative effect on the buffer hit rate, which means that the performance of the database system will suffer. This effect can be resolved by an index reorganization (ALTER INDEX...REBUILD). It is much faster to reorganize an index than to delete and completely rebuild it again. You should run an index reorganization after every larger data archiving session.

If the database system uses a cost-based optimizer (CBO), the statistics will be obsolete after deletion. However, we do not recommend that you run completely new statistics on the optimization of the access paths (UPDATE STATISTICS). Experience has shown that this kind of an update is often not necessary, especially for large database tables. It can even be counterproductive. Generally with smaller data volumes, the optimizer opts for a full table scan, even though a faster index access is possible. The risk that the optimizer takes a wrong decision is less with greater data volumes. It is therefore not necessarily recommendable to update the statistics directly after data archiving, when the data volume is at its lowest.

3.2.6.2 Table Reorganization

Unlike the gaps in the indexes, unfavorable space distribution in data blocks does not have a major effect on performance, because the probability that these data blocks have to be loaded into the data buffer is much smaller than with the indexes. Table reorganizations cost much more than index reorganizations, and should therefore only be used in exceptional cases.

3.2.6.3 Tablespace Reorganization

Although tablespaces may be highly fragmented after archiving, this fragmentation does not affect the performance of a database system. It is therefore not necessary to reorganize tablespaces. However, it may still be beneficial to reorganize your tablespaces to:

- Gain back space from your database system, after you have archived data that is no longer needed and the tables will only be changed slightly from then on. If, however, you expect that the table from which data was archived, will receive the same amount of new data, then it would not make sense to run a tablespace reorganization.
- Make space for other tables. An alternative here would be to accept that after several archiving sessions the gaps will be organized by the free space management function of your database system in such a way that they are large enough to be used for new data, even without a tablespace reorganization.

In sum we can say that to improve the performance of your system, it pays to run an index reorganization after every larger archiving session. When you should run a table or tablespace reorganization you can determine by analyzing the history of how your database blocks are filled on average.

Even without these additional reorganization measures, data archiving stabilizes the size of your database. Even if you need to use several archiving sessions, each data block should at some point in time be completely emptied of old data and at the latest should then be available for new data. Compared to data archiving followed by reorganization measures, this kind of stabilization cannot be felt immediately after archiving and is not noticeable on a higher level.

3.3 Automated Production Operation

In general there are two phases to a data archiving project: An implementation phase in which the initial amount of archivable data is removed from the database and which requires a relatively intensive efforts for analysis, Customizing, and testing; and a phase where data archiving becomes part of the regular processes within the company. In the latter phase the effort for manual administration tasks goes down. A more important focus of this phase is to achieve the most optimal coincidence between archiving processes, system resources and other IT processes, and to try automate the archiving processes as much as possible.

3.3.1 Periodic Archiving

When you schedule a write job, Archive Administration (transaction SARA) checks, whether the variant you have chosen is already being used by other write jobs. If this is the case, it could be an indicator that the same data is unintentionally being archived twice. If, however, you have made certain that the selection criteria and the values of the variant you chose will archive the data that you want, you can ignore this warning. Also, you can schedule the write program to run periodically, by activating the Periodic job indicator when you determine the start data under Date/Time. Archive Administration then periodically generates a
scheduling job, which starts the write program with the same variant without any additional checks. When you choose the period, make sure that for every archiving session the corresponding delete jobs will be completed before the next write job starts.

**Prerequisite for Periodic Archiving**

It would seem that whether or not periodic archiving is possible only depends on the archiving object and the variant. However, periodic archiving cannot be possible if the data to be archived is selected using absolute (document) numbers, items or times. But, if the archivability of the data is determined using specific statuses, which can be set in application-specific Customizing outside the variant, using residence times or other relative time parameters, then the archiving object can be used for periodic job scheduling.

In some cases you can use periodic archiving by leaving out some absolute selection criteria. An example would be the archiving of material documents. Here you do not need to enter the material document number in the variant of the write program (program RM07MARC for archiving object MM_MATBEL). The archivability of the data is determined in the application-specific Customizing of the document runtimes per transaction type. To speed up the selection of the data, however, we recommend that you enter the material document year in the variant. Then you only need to change the variant once a year.

### 3.3.2 Scheduling Data Archiving Jobs

**Job Types**

It is common in data archiving to execute programs in the background ("batch jobs"). Only in the case of analysis programs does Archive Administration provide the option of reading the data in dialogue mode without having to schedule a read program. Depending on the programs that have been assigned to an archiving object, you can have the following types of data archiving jobs:

<table>
<thead>
<tr>
<th>Job ID</th>
<th>Job Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRE</td>
<td>Preprocessing job</td>
</tr>
<tr>
<td>WRI</td>
<td>Write job</td>
</tr>
<tr>
<td>DEL</td>
<td>Delete job</td>
</tr>
<tr>
<td>REA</td>
<td>Read job</td>
</tr>
<tr>
<td>IDX</td>
<td>Index job</td>
</tr>
<tr>
<td>STO</td>
<td>Store job</td>
</tr>
<tr>
<td>RET</td>
<td>Retrieve job</td>
</tr>
<tr>
<td>REL</td>
<td>Reload job</td>
</tr>
<tr>
<td>FIN (END)</td>
<td>Postprocessing job (started automatically)</td>
</tr>
<tr>
<td>SUB</td>
<td>Scheduling job</td>
</tr>
</tbody>
</table>

**Table 1: Data Archiving Jobs**
Figure 13: Job Types in Data Archiving and Their Dependencies

Figure 13 shows the dependencies between archiving object types and the sequence, in which ADK automatically schedules the different job types, if you want to start your programs automatically. The sequence in the Figure refers to the Customizing setting Store Before Deletion. It is of course also possible for the write job to first (or to only) start the delete jobs automatically. The Figure also shows the scheduling options you have from within Archive Administration. When you choose the start date you can choose from the usual options: Immediately, at a specific time, after a specific job or event, with a specific operation mode, or according to a factory calendar.

The scheduling job sees to it that the job it started – usually the write program – contains the job class A. This ensures that the privileged write job can run even on the most restrictively configured database servers. ADK tries to schedule a write job on the database server. This is especially advantageous, when the archive directory is in a file system that is connected locally with a database server. Avoiding network traffic does not only speed up the write phase, which is generally not parallelized, but it also decreases the risk of I/O errors during the creation of archive files.

Use of Server Groups

In the SAP Web Application Server CCMS, you have access to centrally administered server groups for background processing. These server groups are analogous to RFC server groups. This allows you to restrict the distribution of jobs to selected servers with background processes. The introduction of job server groups in CCMS led ADK in SAP Web AS 6.10 to replace the archiving-specific server group concept available with SAP R/3 4.5 with a uniform and extended function. Moreover, you now no longer assign a job server group to individual archiving objects. The job server group assignment now affects all the data archiving jobs of all archiving objects equally. The following general rule still applies: If a group contains a server that runs on the database server, the write job is scheduled on that server.

3.3.3 Interrupting and Continuing the Write Phase

Even if data archiving is usually executed in the background, you may need to interrupt the write phase due to low system resources or other unexpected events. The data objects that have been transferred to ADK up to the point of interruption should be saved in archive files, which should then be properly closed. To ensure that this takes place, ADK offers an interruption concept. Manually or automatically interrupted archiving sessions can be continued by the data archiving administrator at a later point in time, so that they can be completed according to their original selection criteria. For this process to be executed properly, however, the already archived data has to be deleted from the database, meaning that the delete jobs have to run before the interrupted archiving session can be continued. The continued archiving session has the same number as before; it will merely receive the remaining archive files.

You can use automatic interruption if you want to make sure that a reserved amount of disc space is not exceeded, for example. Another motivation for using the automatic interruption function is if you have tight
time windows during which the write phase must take place. You can also interrupt archiving sessions manually from Archive Administration. In Archive Management the interrupted archiving sessions are listed with the stop sign symbol.

Not all archiving objects support the interruption function. You can use transaction AOBJ to find out for which archiving objects interruption is possible. These archiving objects have the indicator *Interruption possible* activated in the detail view.

3.3.4 Options for Automating Dependent Processes

**Event-Driven Deletion**

Data archiving and SAP event control are related in two different ways. On the one hand external processes can control data archiving, for example when delete jobs are dependent on specific system statuses or other processes. This is the case for example, when an employee in the user department checks the detail log generated during the write phase. Only when the responsible user has confirmed that the data selected for archiving was the correct data, can his or her data be deleted from the database. In this kind of scenario it is necessary to implement a release process, which triggers an SAP system or user event. The delete jobs are not started until this event occurs. You can configure event controlled deletion in Archiving-Object-Specific Customizing. You must then enter the event name and if necessary an event parameter in the settings for the delete program.

**ADK-Trigted Events**

On the other hand, data archiving can control related processes. You can, for example, automate the saving of the files: Backup copies of the archive files should be made no later than immediately after the writing of all the archive files. This can be done with special backup software. To run a backup you need a program that feeds your backup software with the names of the archive files that were created during the write phase. This program is scheduled periodically. To do so enter the system event SAP_ARCHIVING_WRITE_FINISHED, which is triggered by ADK at the end of every write phase. If you would rather start the program at the end of the delete phase, enter the system event SAP_ARCHIVING_DELETE_FINISHED. In both cases ADK passes the number of the archiving session as the event parameter. With the help of the function module ARCHIVE_GET_FILES_OF_SESSION an ABAP program can easily determine the physical name of the file.

3.3.5 Controlling Data Archiving Jobs Using an External Job Scheduler

To increase the reliability of data archiving in production mode, you should keep manual interactions to a minimum. This applies particularly to archiving jobs, which can be scheduled outside of Archive Administration. If you are already using a job scheduler for your IT processes, then you will want to use it also to control, optimize and monitor your data archiving sessions. If you are scheduling your data archiving jobs outside of transaction SARA (without automatically starting your jobs immediately through ADK) it is important to keep in mind the following job concepts specific to data archiving:

**3.3.5.1 Scheduling Write Jobs**

The name of the write program for every archiving object has been saved in transaction AOBJ. If you schedule this write program outside of Archive Administration, the following archiving-object-specific checks are not performed:

- Existence of incomplete archiving sessions
- Use of the same variant in other write jobs that have not yet been completed
- Conflict with any future local currency conversions (euro conversion)

In addition you will have to select an appropriate server, a job class, the spool parameters, etc. yourself.

The first two checks in the list help you avoid that you archive the same data twice. If you cannot guarantee that you can prevent this, your scheduler should perform these checks - for check 1, for example, it can call function module ARCHIVE_ADMIN_CHECK_STATUS. The third check on the other hand is not critical, because an additional ADK-internal check takes place at the runtime of the write program. Even externally scheduled jobs that use a different naming convention appear in the Archive Administration job overview. However, they do not appear until they are actually being executed. (For releases below SAP R/3 4.6C see SAP Note 133707.)
3.3.5.2 Scheduling Delete Jobs

- **Direct Scheduling**
  The special aspect of the delete phase is that the number and the names of the archive files that are created during the write phase, are not known before the delete job is started. ADK passes the file information to each automatically started delete job during the write phase. If you determine the name of the delete program using transaction AOBJ and schedule the program directly, files are selected automatically. For a particular archiving object ADK determines the archive file with the smallest key in the status *Writing Complete*. In other words it selects the oldest archive file that contains data still to be deleted. Directly scheduling the delete programs in this way gives the scheduler the greatest flexibility and most complete control over the jobs. The only condition is that only those variants that have been entered in the settings for the delete program should be used for the test and production modes.

- **Indirect Scheduling**
  Figure 14 compares the direct and indirect scheduling of jobs. Indirect scheduling is also available in earlier releases; see SAP Note 205585. Here the scheduler merely starts the program RSARCHD, which then schedules the delete jobs. Which archive files are then processed you mostly decide on your own: You enter the archiving object, the maximum number of files to be selected, and the maximum number of parallel delete jobs, among other information, in the variant for RSARCHD. Based on the sequence and status of the archive files RSARCHD determines the number of delete jobs that are started with each execution of RSARCHD.

![Figure 14: External Scheduling of Delete Jobs](image)

Using the variant you can also determine which of two processing methods RSARCHD uses: Either the program is ended immediately after the delete jobs have been scheduled, or it is ended as soon as all scheduled jobs have the status *Completed* or until one of the delete jobs has the status *Canceled*. The latter method makes it easier for you to monitor your delete jobs, because you would have to look at only the RSARCHD job. If the RSARCHD job finishes properly then the entire delete phase can be considered to be finished properly. If, however, the a job is cancelled, the job status of RSARCHD will reflect this immediately.

4 Storing Archived Data

4.1 Introduction

This chapter deals with the different alternatives for storing and keeping archive files. Archived data must be stored safely and over long periods of time, because in principle it needs to be accessible at all times. Because the data has been removed from the database and now exists exclusively in archive files, the storage security requirements for archive files are very high. During the data transfer, for example, there cannot be any single point of failure (SPOF), meaning that if one component fails, system operations may not be interrupted.

You have the following options to store archive files:

- Storage on an SAP certified storage system or content server, using ArchiveLink
4.2 Storage on an SAP Certified Storage System Using ArchiveLink

You can store archive files on a storage system, using the ArchiveLink interface. You should only use storage systems that have been certified by SAP for ArchiveLink. First, however, we need to define some important terms used in conjunction with ArchiveLink and describe the function of ArchiveLink.

4.2.1 Important Terms in Conjunction with ArchiveLink

4.2.1.1 ArchiveLink, Knowledge Provider, Content Management Service

ArchiveLink is a communication interface between SAP applications and external components, such as an external storage system. It provides SAP applications with a group of interfaces, services and scenarios, with which documents and business processes can be integrated as easily as possible. ArchiveLink also contains an interface to storage systems. The interface makes the functions of the external systems available in the SAP system. ArchiveLink includes a comprehensive display function and its own document viewer for accessing and displaying documents. ArchiveLink comprises the following technological components:

Figure 15: Integration and Components of ArchiveLink

Up to and including SAP R/3 4.5A only ArchiveLink has met most of the named functions requirements for SAP applications. An SAP application integrates ArchiveLink functions and in this way can access the administrative and interface functions, to process documents on external content servers. Particularly important with ArchiveLink is that it operates with very simple and small management structures and application interfaces. The advantages of this kind of architecture particularly lie in the fact that they are easy to integrate into business applications and can handle mass data due to their slim management tables.

As of SAP R/3 4.5B this concept was considerably expanded in terms of the functions offered by the management layer, with the development of the Knowledge Provider (KPro). KPro offers an application interface that is based on content models, which allow for a very comprehensive and flexible modeling of documents. This provides the applications with a large number of functions that go beyond those offered by ArchiveLink, such as versioning, variant creation, indexing based on content models, and many more. As a result, SAP applications can, depending on their document requirements, use KPro to integrate any given number of data management functions, or to simply use ArchiveLink.

In sum, ArchiveLink and KPro differ mainly on the application interface and management layers. The content server interface is the same in both components, so that the same document functions can be used on the same external content server.
The components we mentioned are generally named as follows:

- **External Content Server**
  In connection with KPro, you talk about content servers which have content repositories. As of SAP R/3 4.5B these terms are also used in connection with ArchiveLink. In addition, the terms "storage system", "archive system" and "external archive system" are also used as synonyms.

- **Content Server Interface**
  As of SAP R/3 4.5B the terms “ArchiveLink Interface” and “ArchiveLink HTTP Content Server Interface” are used. The “HTTP Content Server Interface”, with a slightly less comprehensive array of features, was also released with SAP R/3 4.5B.

In sum we can say the following about the array of components and terms we have mentioned:

- All functions that are used for the integration of documents in SAP applications, use the ArchiveLink interface to certified storage systems to store documents.
- This way the same storage systems can be used for all application purposes, regardless of whether the application uses an ArchiveLink or KPro application interface.
- The terms “ArchiveLink Interface” and "KPro CMS" can be used interchangeably and refer to the use of document-level application interfaces. This is especially important for data archiving because as of R/3 4.6C it uses the KPro CMS Application Interface instead of the ArchiveLink Application Interface, for reasons of simplicity.
- Generally, in this document we will use the term "ArchiveLink", which is also the most commonly used term among SAP users. This term originates from the well-established ArchiveLink Interface. However, at some points in this document we do distinguish between ArchiveLink, KPro, and CMS, for the sake of precision.

4.2.1.2 The Concept of “Document”

Before we can talk about the purpose of ArchiveLink, we must first define the term "document" in the context of ArchiveLink, from both a business and a technical point of view.

ArchiveLink distinguishes between four different document categories:

- **Incoming Documents**
  Can trigger business processes; from a technical standpoint this category comprises every type of document that was not created by the SAP system itself, such as scanned documents or files on your local PC. ArchiveLink passes these documents unchanged from their original format on for storage.

- **Outgoing Documents**
  These are created as part of a business process or as the result of a business process; from a technical point of view, this category comprises documents that were created with the help of SAP text processing systems SAPscript or SmartForms, and are output using, for example, a printer, FAX or monitor. ArchiveLink is used to pass outgoing documents to the storage system in PDF format.

- **Print Lists**
  These documents are created during the evaluation of business processes; from a technical point of view they are the result of the evaluation program. They are transferred to the storage system via ArchiveLink as ASCII files. For more information, see Chapter 1.3.2.

- **Archive Files**
  These contain business process information that is removed from the SAP system database in the form of data, which must remain accessible in the future. They can be displayed directly as documents.

From a technical point of view the last three categories produced within the SAP system.

4.2.2 The Function of Archive Link

ArchiveLink facilitates the automatic integration of business documents into SAP applications. Originals, such as scanned incoming invoices, are linked to the corresponding business objects, consisting of application data, and as a result are accessible directly from the business process or the application.
document they are linked to. The ArchiveLink concept goes further and facilitates the integration of documents with business objects independent from how the documents are physically stored on external storage systems.

The central function of ArchiveLink is the integration of all objects and documents that belong to a business process, which creates the following advantages:

- It allows you to use all SAP search options for documents, because the stored documents are accessed from the SAP application document.
- The documents are integrated into the business processes because they are linked to their corresponding business objects.
- Data is protected through the SAP authorization concept (application authorizations, ArchiveLink authorizations).

Documents managed with ArchiveLink are physically stored on external storage systems or content servers via the ArchiveLink interface developed and defined by SAP. ArchiveLink connects SAP applications with defined external storage systems. ArchiveLink is used to control the different components of the external storage system from within the SAP application processes.

A typical configuration of an ArchiveLink system comprises the SAP system, an external storage system for storing unchangeable documents and possibly an SAP content server for storing documents that are still being modified (changeable). On the client side you have centralized or decentralized data entry stations (for example for the scanning and storing of documents) and clients, which are used to access the documents.

![Figure 16: Typical Configuration of an ArchiveLink System](image)

4.2.2.1 Communication Using ArchiveLink

The ArchiveLink interface consists of both server and client components. A typical call of the server interface would be triggered when a file originating in the SAP system is to be passed to the storage system. An example on the client side would be the request for the display of a document.

The ArchiveLink interface is available in four versions, for which a third party vendor can be certified. The versions are named after the SAP R/3 release with which the enhancements were first made available. (The interfaces are available to all SAP solutions.)

- **ArchiveLink 2.1**
  
  First version of the ArchiveLink interface. The server communication is based on remote procedure calls, while the client integration takes place via the ArchiveLink Viewer.
• **ArchiveLink 3.0**  
  Server communication now takes place via SAP RFC, and on the client side OLE automation 2.0 is now supported.

• **ArchiveLink 3.1**  
  The interface is enhanced with more functions.

• **ArchiveLink 4.5**  
  The interface is expanded to HTTP. RFC is eliminated and the ArchiveLink Viewer is no longer part of the interface.

The current SAP systems support the last two versions of ArchiveLink, meaning 3.1 and 4.5.

The HTTP Content Server Interface 4.5 is closely related to the ArchiveLink Interface 4.5. It contains all HTTP components of the ArchiveLink Interface 4.5, but does not include OLE automation and the barcode BAPI. This means that a system that is certified for the HTTP Content Server Interface 4.5 cannot be used for all ArchiveLink scenarios. Therefore, almost all vendors are certified for ArchiveLink Interface 4.5.

### 4.2.3 Storing Archive Files

The communication between the SAP system and the storage system during the storage of archive files can take place synchronously or asynchronously, depending on the SAP R/3 release you have implemented and the ArchiveLink interface you are using. The difference between asynchronous and synchronous only refers to the interface. In SAP systems the storage of archive files always takes place asynchronously, either through background jobs or through an asynchronous job management process. The following table gives an overview of the different options available for storing a file:

<table>
<thead>
<tr>
<th>SAP R/3 Release</th>
<th>ArchiveLink Interface Version</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3.1 (RFC)</td>
</tr>
<tr>
<td>4.0</td>
<td>asynchronous</td>
</tr>
<tr>
<td>4.5</td>
<td>asynchronous</td>
</tr>
<tr>
<td>4.6</td>
<td>asynchronous</td>
</tr>
<tr>
<td>4.6C</td>
<td>synchronous</td>
</tr>
<tr>
<td>Application systems that are based on SAP Web Application Server (for example SAP R/3 Enterprise)</td>
<td>synchronous</td>
</tr>
</tbody>
</table>

Table 2: Storage Methods Via ArchiveLink

The HTTP interface only supports synchronous communication. Therefore, the storage procedure for archive files takes place synchronously. The RFC interface (ArchiveLink 3.1) supports both asynchronous and synchronous methods. In SAP R/3 Releases up to 4.6 the storage (and retrieval) of archive files takes place asynchronously. As of SAP R/3 4.6C synchronous communication is used.

### 4.2.4 Accessing Archive Files

Archive files stored using ArchiveLink are accessed directly in the storage area or they are first copied back from the storage area into the file system of the SAP system. The actual information of the archive file is then accessed using this local copy of the archive file.

There are two different methods used for accessing the information:

• **Accessing an archive file block specified through a position and an offset**  
  This is the case with single document accesses and evaluations directly in the storage area. A single document access requires few block accesses on the storage area, and during an evaluation the entire archive file is called sequentially in blocks. This may mean a large number of requests, which, however, each involve small data volumes.
• **Accessing the entire file**
  This is the case, for example, when the file is copied into the local SAP file system. The function is executed manually by the administration on the level of administration sessions. This means that the number of parallel retrieval requests to the storage area is relatively low, but that each request involves a large amount of data.

The method of accessing archive files can be synchronous or asynchronous, depending on the release and the interface version. The table is only valid for the function of copying entire archive files into to SAP file system, not for block accesses to data in the storage area. Block accesses via a position and offset are always synchronous.

In sum, the storage system you are using should meet the following requirements:
- It should support a high number of synchronous parallel requests.
- These requests should cover small and very large data areas.

### 4.2.5 Certified Systems and SAP Content Server

In addition to the certified partner systems, SAP offers its own SAP Content Server as of SAP R/3 4.6B. This content server also supports HTTP Content Server Interface 4.5.

The SAP Content Server should not be confused with an archive system. It is an independent windows-based system, which is connected to an SAP database (SAP DB), in which the documents are stored. The SAP Content Server is used for documents that are delivered by SAP, such as via the SAP Knowledge Warehouse, and for documents that are created through defined processes (attachments in the Workflow). In addition it is used extensively in SAP Document Management (DMS) for documents that are still being processed. The SAP Content Server may not be used as an archive system for storing documents that can no longer be changed and that must be kept by law, because it saves data in the database and does not support any other media. In the ArchiveLink environment it is used as a document cache server in distributed systems.

### 4.2.6 Searching for Documents

Users usually access stored documents directly from their application transaction. It must also be possible for the user to access all documents that are linked through a business process, and for him or her to have an integrated access to SAP documents and non-SAP documents. Documents can be linked to several business objects of a business process, including to documents that have been stored on an external content server via ArchiveLink.

The following application components provide an integrated view of all documents in the SAP system:

- **Document Finder**
  Allows you to search for and access documents using different criteria. The Document Finder can search for documents regardless of whether the search properties are being managed within or outside of the SAP system.

- **Records Management**
  This tool goes far beyond the document approach to searching, and integrates different information in electronic files, regardless of whether the searched for item is a document, business object, transaction, program, workflow, or other; also it does not matter whether the individual information units are being managed in a single SAP system, are distributed across several SAP systems, or are managed in non-SAP systems.

### 4.3 Storage Using an HSM System

#### 4.3.1 What is HSM?

HSM means Hierarchical Storage Management. HSM systems integrate several different storage systems and organize the physical distribution of the files across these media, based on varying, individually
customizable rules. A simple rule is, for example, the automatic move of files from fast to slow media, depending on the frequency with which this data is accessed.

**Figure 17: Hierarchical Storage Management System (HSM)**

HSM systems can be accessed as if they were file systems, meaning that you do not need any special interface to be able to use an HSM system for writing or reading data. The HSM software takes over the automatic conversion of the file and folder paths, according to the predefined rules. As a result an HSM system can be accessed by any system as if it were a hard disk of potentially unlimited size. Through the transparent integration of different storage media and architecture, the HSM system presents itself as a single file system.

The concept of an HSM fulfills two main objectives:

- Offering direct, standardized access to a storage pool that can always be expanded. The expansion takes place only in the HSM system. The application that is accessing the data is not affected by this expansion and does not have to be switched to a different file system or interface.
- Intelligent storage through intelligent use of different media, which helps users save on costs.

The potential cost savings generated by HSM systems can best be demonstrated using an example:

Within a given application data is created that needs to be accessed often and other is created that needs to be accessed less often. For example, an invoice booked in SAP R/3 Enterprise is accessed most often during the year it is first posted. At the end of that year the information is only needed for evaluation purposes. In the following years it is needed only in exceptional cases, for example for audits or test purposes.

During the time the document is accessed frequently the data must be available online at all times, meaning that it must be either in the database or in archive files that can be accessed directly and quickly. Data that is accessed infrequently can also be stored on slower, possibly offline media, which is less costly. It is exactly this aspect that HSM systems aim to satisfy through the integration of different storage systems and the use of definable migration rules. As a result, data is migrated automatically and completely transparently for the user. It is not necessary for the data to be copied manually from one medium to another.

A common architecture is a combination of hard disc, jukeboxes with optical discs and tape robots. In this scenario one rule could be that files, which are accessed regularly, are stored directly on the hard disk. If the access frequency falls below a certain threshold, the files are automatically transferred to optical discs; if the frequency falls below another threshold the data is moved to magnetic tapes.
4.4 Manual Storage

For the manual or alternative storage of archive files it makes sense to use or reuse storage systems that already exist in a company. Then you do not need to make any additional investments in hard or software. Usually the type of storage systems used would be large robot systems or jukeboxes, that are already being used in a company. These storage systems can be integrated into the data archiving process either directly or indirectly.

4.4.1 Direct Integration

Most jukeboxes have their own file system interface, for transferring archive files to jukeboxes. For data archiving this option can be used as if it were an HSM system, meaning that the data is directly written to the file system of the jukebox.

If you want to use this scenario you must make sure that the archive files are directly written to the safe medium. At the very latest the request close archive file at end of write program must ensure that the file is saved instead of only existing in the cache area. If this function is not supported by the file system driver of the jukebox vendor, you should not implement this scenario without additional safety measures.

4.4.2 Indirect Integration

In this scenario the storage system is used after archiving as a backup medium. The archive file is created in the file system of the SAP system and is saved via administration before the delete phase.

4.4.3 Advantages and Disadvantages of Manual Storage

The main advantage of manual storage is cost savings, because you do not need to incur any additional purchase costs. Another advantage is that you can reuse existing processes, which can be easily expanded to include additional backups of archive files.

The central weakness of manual storage lies in the fact that it requires additional administrative efforts, unlike the automated processes of ArchiveLink and HSM. This would have to be analyzed on an individual basis.

The advantages discussed under ArchiveLink and HSM do not apply here and can be achieved either with a lot of effort, or not at all. Whether or not a solution meets the requirements for security, performance, costs and durability, must be decided on an individual basis. The responsibility for the solution lies entirely with the company that has implemented the solution. There is no system vendor that could take responsibility for possible problems.

4.4.4 Criteria for Choosing a Storage Strategy

Before you decide on a specific storage strategy or a specific storage system, you should first analyze the most important selection criteria. However, keep in mind that there is no such thing as a "best" solution that meets 100% of all criteria. Therefore, you must adapt the following list of criteria to your specific needs. This is particularly important, because some criteria can contradict each other. For security reasons it may make sense to store your archive files multiple times on low-level storage media, before the delete phase.

However, if your aim is to ease the load on your database, then this strategy may be counterproductive, because the data must be removed from the already overloaded production database. It does not make any sense to have higher requirements for your storage system than for your leading SAP system. For example, a catastrophe case scenario would only be relevant if it is also taken into account in the SAP system.

The list of criteria discussed below is not meant to be an exhaustive list. The list is merely meant to name a few of the criteria that could come into play when you choose your storage strategy. Each storage strategy should be analyzed based on the specific needs of a company and taking all criteria into account.

4.4.4.1 Security

Under the criterion security you should pose especially the following questions:

- How secure is the path to the storage system?
This question applies to the entire process – starting with the creation of the archive files up to their final storage on storage media outside the database. It is essential that the created archive files are transferred to their final storage destination unchanged. This is especially important when archive files are moved from one system to another. The process must ensure that the correct data is transferred to the final storage medium.

- **How safe is the storage medium?**
  How secure a medium is depends largely on the type of medium being used, such as hard disc, MO, WORM, CD or magnetic tape. These can differ considerably in terms of security, performance and lifetime, and this must be taken into account in the purchase decision.

- **Which backup options does the storage system offer?**
  The storage solution itself, which generally also includes its own database, must offer an adequate concept for securing the data that is stored on it. Also, the backup options it provides must be integrable into your current system. If not, the implementation of your storage system will generate additional costs, because you will need to maintain and administer different backup systems.

4.4.4.2 Costs
The costs of storage scenarios can vary considerably. Not only do the products differ in terms of their initial costs, but also in terms of the follow-on costs they generate in the company. These include money spent on support, training and knowledge transfer, as well as the expenses for having to adapt internal IT infrastructures if necessary. In a long-term view it is more important to look at the continuous operating costs instead of the initial costs, as the central cost component of a new storage solution.

4.4.4.3 Integration
A new storage system should be integrable into an existing IT landscape both from a technical and an administrative point of view. Here it is also important to take the costs such an integration would generate into account.

4.4.4.4 Additional usage
You should ask the question for what other purposes your new storage system could be used, besides for the storage of archive files. Usually storage systems are also used in the context of document management and data backup. Many systems used to store archive files offer other comprehensive features for document storage and management. Even though when you first implement a storage system your main focus may be data archiving, you should check, whether or not your company is planning to implement a document management or document integration strategy in the future.

You can also use your data archiving storage system for backup purposes for data from other areas of your business. In sum a storage system can be useful for storing archive files, and also for backing up other file systems or databases in your company.

4.4.4.5 Performance
From the point of view of the SAP system the purpose of the storage system is to make the stored data available to programs that want to access this data. This can encompass a sequential access to the entire archive file or a direct access to a single business object, whose location within the archive file is specified by the byte position and length of the used up space. In addition, it must support that archive files are copied completely to or are removed completely from the storage area. Generally these type of operations are possible with all types of storage systems: In the case of an HSM system you would use the corresponding system commands, and in the case of storage systems certified for SAP ArchiveLink through the requirements of the certification.

4.4.4.6 Long-term storage
Especially when you store archive files you must make sure that the storage system you use is suitable for long-term storage. The data stored there must be accessible over several years.

Statistically archive files are accessed less and less frequently as time goes by. The less an archive file needs to be accessed, the lower the requirements for access time, until eventually the files can be stored offline, to be accessed only via manual administration. As access time becomes less and less important, because data is accessed only seldomly, you can also lower the costs of your data storage, by using storage hierarchies. The data that is accessed infrequently or not at all can be moved to slower, but less costly data carriers. To be able to use this option your storage system must support this storage technology and provide
time or data controlled automatic mechanisms for the migration of data from fast and expensive to slow and less costly media.

4.4.5 ICC: Premium Integration Service

SAP offers third party storage vendors a premium integration and certification service (ICC) for the certification of their interfaces. ICC also offers other services, such as the use of test systems. A new part of the service is an expanded certification program to optimize the integration of third party products into SAP software. To be able to use this service vendors must have obtained the initial certification for their interface.

Part of the extended service offerings is the AL-LOAD-Load Test which is made up of two phases:

1. Write phase
   a. Large amounts of data are sent to the archive system.
   b. The time needed for the archiving session is recorded.

2. Read phase
   a. The selected archive files are called up.
   b. The time needed for the request is recorded.
   c. The retrieved documents are checked for correctness.

The load test guarantees a higher degree of integration of interface products, which increases customer satisfaction. For more information about the certification and integration service see http://www.sap.com/ICC in SAPNet.

5 Access to Archived Data

5.1 Introduction

There are a variety of different options for accessing archived data. In principle it is possible to read every archive file that has been created in the same system and client from which it is being read. The exact characteristics of a read program of a specific archiving object depends on the programs provided by the applications. The array of possibilities is very wide ranging: At the lower end of the spectrum you have applications that do not supply any special programs for accessing archived data. In these cases the archived data can only be displayed using the Archive Information System. However, this type of display is very technical, much like the data browser display (transaction SE16) for data from the database. At the other end of the spectrum the archive access is integrated into the application so well that the end user does not notice whether the displayed data is in the database or in the archive.

There are different scenarios available for accessing archived data:

- **Archiving data completely**
  Many companies have data that they know will not need to be accessed again. This data can be completely archived and stored in archive files using SAP Data Archiving. The data will never have to be accessed again.

- **Print list storage**
  If before you archive your data you already know what kind of data evaluations you need to perform in the future, you can run these analyses before archiving and store the resulting print lists on the appropriate storage media. When you need to access your data later, you can call up the corresponding print list. Here, however, the archive is not actually accessed.

- **Index using the Archive Information System**
  If the archived data needs to be accessed later, you can use the archive information system to search for, format, and display the data. For more information, see Chapter 5.4.

- **Document Relationship Browser**
The Document Relationship Browser (DRB, Transaction AL01, User Role SAP_DRB) is based on the Archive Information System. You can use it to display the relationships between data within a business process or even across different business processes (see Chapter 5.5).

- **Display from the application**
  For some archiving objects you can display the archived data directly from the application in a business view. In this display you cannot see whether the data being displayed has been archived or whether it still resides in the database. The only difference is that the archived data can no longer be changed.

The following chapter focuses on archive accesses using the Archive Information System and the Document Relationship Browser. We will also describe the use of sequential read programs. We will not go into further detail about print list storage, because it is only slightly related to the other topics.

### 5.2 The Fundamentals

Independent of the type of access you will use, the following steps are necessary for the archived data to be identified and displayed. It is mainly in the implementation of these steps that the various access options differ.

1. **Selecting the archive files and the business objects to be read within an archive file**
   Here you can choose from two different techniques:
   - In the first technique the data is selected manually by the user. The selection of the archive files takes place in a selection screen offered by the system.
   - In the second technique the archive files to be read are selected by the system without the need for any further user interaction. This is done using an archive index, which the system reads based on the selection criteria entered by the user. An archive index is a database table that contains the application-specific selection fields, such as document number, and the key of the archive file in which the requested data is located.

2. **Opening the archive files and reading their content**
   There are also two options for carrying out this step: The first is to open the archive file and to read the contents of the archive file sequentially. The second is a direct access: In this option the file pointer is positioned inside the archive file directly where the business object to be read starts. This place is called the offset.

3. **Filtering the desired data**
   The selection criteria with which data is to be read from the archive generally does not coincide with the selection criteria used to archive the data. This means that by selecting the archive files to access the desired data, more data than you want is actually read from the archive. Therefore the program must filter the data the user asked for out of the selection, even if it has already read the business objects in question.

4. **Displaying the desired data**
   The data that is to be displayed from the archive can appear in a variety of formats. The spectrum of options ranges from a purely technical view to a business view usually used to display data from the database. The first option is available in the archive information system, while the second option is available in applications that have integrated the archive access completely in their display function.

### 5.3 Sequential Read Programs

Users working with data archiving usually notice the Read function in Archive Administration (transaction SARA) first in the context of archive accesses. This function usually has programs to sequentially read the archive files the user has chosen. These programs were written especially for evaluating archive files and generally operate exclusively on archived data. The data is usually displayed in a format that has been tailored to the needs of the end user.

An example of a sequential read program is the program RKAARCS1, which belongs to the archiving object CO_ORDER (internal orders). This read program is available via the Read
function in Archive Administration. After entering the selection criteria you can execute the program which brings up the dialog box for selecting the archive files. Keep in mind that the selection criteria do not influence which archive files are offered for selection. You will always be offered all accessible archive files for your analysis, no matter what your selection criteria. You should therefore make sure that the selection of archive files matches your selection criteria. Do not choose all relevant files, because this way not all of the data you want to see will necessarily be displayed. If you choose too many archive files, you may be facing a long response time, because the program reads all files through sequentially.

As a next step the read program reads the selected archive files sequentially and filters the data according to the selection criteria you entered. The selection criteria do not affect the runtime of the program. The runtime is affected by the selection of the archive files. The contents of the archive file is usually displayed as a list. With internal orders you have the option to use this list to navigate to more detailed information. However, this is not very typical for this type of analysis.

You can run a read program in the dialog mode or in the background. The scheduling of the program is similar to that of a delete program. The difference is that the read program requires a variant for the transfer of the selection criteria.

The programs available in archive administration are usually dedicated archive read programs, you can also have programs that were originally developed for evaluations of database data, but were later extended to also perform archive accesses. A certain disadvantage of these programs is that users must know, whether or not the data is in the archive, and if so, in which archive file it is stored. The advantage is, however, that the data is displayed in a format that you are familiar with. An example of this are summary reports (Report Writer Reports) in overhead cost controlling.

Use the function Data Source in the selection screen of this type of read program, to indicate that the data should be read from the archive. Here you also choose the archive file.

From a technical standpoint the selection of the data source (database or archive) and the archive file to be read belongs to the selection screen, although the information for the data source selection is not seen directly on the selection screen. This means that when you save a selection variant the data source is also saved with it. This allows you, for example, to create a variant to select specific archives in the background. In the list that is displayed after you have executed the program, you cannot tell whether the data being displayed is in the database or in the archive.

5.4 Archive Information System

This tool was especially designed for archive accesses, and it allows you to configure archive indexes, fill these indexes with data from the archive and search for archived data. As in an application-specific archive index, archive file and offset are added, which allows you to directly access your data. In addition, the Archive Information System offers a generic view – not application specific – of all contents of a business object from the archive. This works for all archiving objects, even customer-specific ones and requires no application-specific programs or modifications.

The central concept within the Archive Information System is the archive information structure. (For simplicity’s sake we will use the commonly used short form "infostructure" from here on forward.) In a certain manner this term replaces the above-mentioned term archive index. Each archive access performed from the Archive Information System takes place via an infostructure. SAP delivers standard infostructures, which usually cover most customer requirements. It is also possible, however, to adapt the infostructures or to create customer-specific infostructures with the help of the Archive Retrieval Configurator (Customizing component of the Archive Information System); see Chapter 5.4.1 Creating an Infostructure. As the archive index, the infostructure is filled with data either directly during the delete phase or subsequently by the user. Also similar to an archive index, the data of an infostructure is kept in a database table. Another component of the Archive Information System is the Archive Explorer. It is used to search for data within an infostructure and allows you to access and display archived data directly.

Each infostructure belongs to exactly one archiving object and also one field catalog. A field catalog is a collection of fields, which are used to index the archive files of a specific archiving object. The fields of an infostructure are always a selection of the fields of the corresponding field catalog. The field catalog also contains a series of technical properties that are also incorporated into the infostructure. Because of the field catalogs, it is not necessary for you to know the technical properties of your archiving object to create an
infostructure. These properties are already contained in the field catalog. To create an infostructure you only need to select the fields from the field catalog.

**Archiving Object**

**Field Catalog**

**Archive Infostructure**

![Diagram](image)

**Figure 18: Field Selection in the Field Catalog**

In the following section we describe the different uses of the Archive Information System and provide some background to these uses. The steps are listed in the order in which a user or an administrator would normally perform them, when the Archive Information System is used for the first time for an archiving object. These functions are available through the central administration of the Archive Information System (Transaction SARI). For more information about the Archive Information System see the Application Help in the SAP Library.

5.4.1 Creating an Infostructure

The Archive Information System comes with standard infostructures, which usually meet all the requirements of a user for analyzing archived data. If this is not the case, you can also copy a standard infostructure and adapt it to your specific needs. You should not, however, make any changes to SAP standard infostructures. Any changes of this kind are considered to be modifications and may be overwritten with the next upgrade or support package installation.

When you create an infostructure you determine which fields from the archive will be incorporated into the infostructure. To do this you choose the fields you want from a field catalog and incorporate them into the infostructure. For many archiving objects field catalogs already exist as part of the standard. In exceptional cases, if no field catalog meets your requirements, you can create your own.

For technical reasons some fields of the field catalog are directly incorporated into the infostructure and cannot be removed. These are generally the key fields. Most of the fields of a field catalog, however, are part of the group of selectable fields, and can be chosen for the infostructure. Later you can use all fields in the infostructure to search for archived data. Keep in mind, however, that the infostructures are saved in the database and that therefore every additional field incorporated into the infostructure takes up additional space in the database.

5.4.2 Activating an Infostructure

To be able to use an infostructure you must activate it. An infostructure can only be filled with data from the archive and evaluated if it has been activated. Active infostructures can no longer be changed.

In addition an evaluation program is generated for the evaluation of the corresponding table and for the archive access to display archived data. After the database table and the evaluation program are generated, the system marks the corresponding infostructure as active. This indicator means that the infostructure can
now be used for evaluations and that it will be built automatically during the execution of the corresponding delete program.

5.4.3 Building an Infostructure

5.4.3.1 Automatic

During the delete phase all active infostructures belonging to an archiving object are filled automatically with the data of the corresponding archive file. Based on the defined infostructure the Archive Information System filters the data from the data records in the archive and inserts them together with the key of the archive file and the offset of the business object into the generated database table. These entries are then the basis for later data searches.

5.4.3.2 Manual

An infostructure can also be filled later for already existing archives. The manual option allows you to build infostructures to evaluate data that was archived before you implemented your Archive Information System, or after you have changed the fields of an infostructure.

When you build an infostructure the generated database table is filled with data from the archive and a fill status log is kept. In status management of the Archive Information System you can check these statuses to see for which archive files corresponding infostructures have been built.

5.4.4 Displaying an Infostructure

The search for archived data in Archive Explorer is always conducted using the analysis program that is generated when the infostructure is activated. The selection screen of the evaluation program contains all fields of the infostructure, except the client, archive file key and offset fields. When you execute the program you receive a list of all entries in the infostructure that match your selection criteria. Up to this point in the evaluation process the archive has not been accessed. The system has only read the index data that was saved in the infostructure. You can double click on an entry in the list to access the archive directly and navigate in the data hierarchy down to field level.

The view of the data shown by Archive Explorer is technical and therefore more suitable for power user and administrators. The Archive Information System provides a technical view for every archiving object.

To use the normal view of an infostructure using Archive Explorer, the infostructure must have been built previously. This means that only the data that has the status Deletion Complete, can be evaluated. This also makes sense, because all other data is still in the database and there is no reason to look for this data in the archive. It is also possible for you to check over the data that was archived before you start the delete phase. You can do so using the function Ad-hoc Evaluation. With this type of evaluation the system does not access the generated database table, but carries out a sequential read access on the archive files you have chosen. The data set, which would be created when you build the infostructure, would only be saved internally. The subsequent display of data and the navigation options available will then correspond to those of a normal infostructure evaluation.

The evaluation of built infostructures using Archive Explorer, or other accesses to the Archive Information System, are especially fast if the system can make the access via the primary index of the corresponding database table. For more information about this topic see SAP Note 164704.

5.4.5 Deleting an Infostructure

You always have the option of manually deleting infostructures. This function gives you the additional flexibility to build or delete infostructures as needed, for example when you do not constantly need to access the archive. The data saved in a generated database table use up disc space. For this reason it generally makes sense to delete data from older archive files, after a certain amount of time has passed.

In contrast to the building function for archive infostructures, the delete function is not integrated into ADK. This means that the deletion of infostructures has to be started manually. This is particularly important when you reload archives. When you reload archived data (which should only be done in exceptional cases), the active infostructures of the data in question have to be manually deleted and other infostructures have to be manually built for archive files that may have been created during the reload.
5.4.6 Using Display Variants

To avoid unnecessarily using up database space unnecessarily, we recommend that you refrain from using several infostructures for one archiving object. This generally leads to redundancies of data, which should be avoided. As an alternative, you can create an infostructure that contains all the fields you need to cover your different display scenarios. You can then create several display variants that contain only those fields that you need for a specific scenario. This way the fields of the infostructure only exist once.

5.5 Document Relationship Browser

The Document Relationship Browser (DRB) is used to show the relationships between business objects and the business objects themselves. Generally these are documents that were created within one business process or that belong to the same process. DRB shows linked documents from different application areas and across systems, including archived objects. In this chapter we will particularly focus on DRB features that have to do with data archiving. For more information about this topic see SAP Note 492938.

5.5.1 DRB and Data Archiving

DRB is not an independent application, but a service that is called for an entry object. The applications have different transactions and programs from which you can branch to DRB for a specific entry object. Most of the functions are grouped in the role Document Relationship Browser (SAP_DRB). For some business objects, such as sales orders or invoices, it is possible to directly branch to the DRB from the Archive Information System. In addition you can branch directly to DRB from some document display transactions, such as VL03N for deliveries, via Environment \rightarrow Relationship Browser.

After entering DRB via a business object of a specific type, the program shows which business objects are linked to this entry object. The applications deliver the objects that are directly linked to this entry object. What this means in more detail depends on each application. The links between the business objects do not have any further semantic, meaning that you cannot tell whether an object is the predecessor or the successor of another object.

To avoid a cyclical and therefore unnecessarily complicated representation of linked business objects, each object is only displayed once within the business process. As a result, not all direct links are actually displayed. Also, the representation may vary. It is certain, however, that the number of displayed objects remains the same, independent of the sequence of the individual navigational steps. In the DRB view only those objects are displayed that are directly linked to the entry object. If other objects are linked to these objects, you can display them by navigating in the link tree. Double click on an object key to display an object. What this display looks like depends on the application in question and the type of business object.

As mentioned, DRB needs the support of the application, for example to find the entry object the user selected. This can be resolved with the functions of the Document Relationship Browser role (SAP_DRB). Many of the functions in this user role include automatic access to the archive. For Releases 3.1I-4.6C you can also use ALO1 to determine document relationships.

The way a specific business object is displayed in the DRB view also depends on the application. Also whether or not an archived object is displayed differently from a like object in the database, depends on the application and on the object type.

5.5.2 Searching Options

The following items appear as radio buttons on the selection screen and are used to determine where the search for a document is to take place:

- **Search in DB**
  When you select this option the program only searches for the documents in the database. Archived documents are completely ignored.

- **Search in DB and SAP AS**
  If you select this option the program searches for the documents in the database and in the above-mentioned infostructures of the Archive Information System. The archive, however, is not accessed. As a
result, not all fields may be filled in the results list, or not all desired records can be found, because the program views fields that are not contained in the infostructure as empty and skips them.

- **Search in DB, SAP AS and Archive**

  When you select this option the program searches for the documents in the database and in the Archive Information System. For documents that were found according to the Archive Information System, the data that may still be missing is searched for in the archive. This means that here, too, only documents are read that are in an appropriate infostructure.

  This selection option only determines what will be displayed in the results list of the program and not which linked documents DRB eventually finds. It is therefore possible that although you selected the option Search in DB, archived data will be displayed in DRB as linked objects. In many cases only the two options Search in DB and Search in DB and SAP AS can bring back meaningful results if used. The option Search in DB and SAP AS is often faster than the latter option, but often times leads to confusing results, because the end user usually does not know which fields are contained in the infostructure and which effects this would have on the selection.

Unlike in financial accounting, in logistics DRB does not display archived documents in the same way as database documents. However, the display transactions for archived documents were designed to be similar to the corresponding display transactions for documents still in the database. In addition, all the important fields are displayed. If the documents are still in the database the usual document display transactions, such as VA03, are used.

All other logistics object types are connected to DRB in a manner similar to sales orders. The only differences are in the case of the field catalogs used, and in the case of those fields that can be used to make selections and that can be integrated into the infostructures. For more information see the documentation for the application-specific components of the Document Relationship Browser.

### 5.5.3 Configuring and Personalizing DRB

To optimize access to archived data and to be better able to adapt the DRB functions to the end user, DRB offers the following configuration options:

- Presetting the entry programs
- Choosing entry list fields
- Choosing object types to be displayed
- Choosing fields in DRB
- Improving performance

In general many settings can be user-specific. In fact, all settings, except for the selection of the object types to be displayed, are not specific to the Document Relationship Browser, but originate from the tools used in it. However, since these settings are extremely useful for adapting DRB to data archiving, we will now discuss how you can make the access to archived data even more convenient for the user.

#### 5.5.3.1 Presetting the Entry Programs

The standard role “Document Relationship Browser” contains several programs that can be used to access DRB; however, these programs are configured in such a way that they do not make any archive accesses. For the logistics programs the search option Search in DB is set as a default. In the accounting programs the automatic access to table ASACCESS01 is switched off by default. In this section we describe how you can assign these programs to a role in such a way that the archive is accessed automatically.

First you must create a selection variant for every program you want to use. You can use the field properties of the selection variant to preset and hide the Search in field. If you start the program with this variant the user will not see these fields anymore on the selection screen and the desired value is used automatically.

You can do the same for the entry lists of accounting documents and line item reports in cost accounting. However, here it is not possible to hide the fields for the data source selection, because either way they do not appear on the selection screen. They are saved together with the variant. Of course you can also control the entry lists for accounting and cost accounting documents via table ASACCESS01, as mentioned above. However, this would mean that all users are affected by any changes. If you really want to configure the
system in such a way that the cost accounting line item reports automatically read from the archive for all users, then it would be preferable to make your settings via table ASACCESS01.

After you have created variants for all programs you want to use, you can enter these programs in a role (transaction PFCG). If this kind of a program is called from the role it was assigned to, it starts automatically with the presets of the variant. This way you can create a role that contains all programs that lead to the DRB, and which are configured in such a way that they automatically access the archive. You can also use this mechanism to pre-enter other selection criteria than those we have named.

5.5.3.2 Choosing Entry List Fields

Many of the programs contained in the role “Document Relationship Browser”, were created with the help of the ABAP List Viewer. Whenever a list is displayed, you can change the layout, save the changed layout and set a layout as default. These settings can be user-specific or can be valid for all users.

5.5.3.3 Choosing Object Types To Be Displayed

Usually the representation of complex business events and processes is also complex in DRB. Also, due to the large number of object types that DRB supports in SAP R/3 Enterprise, you could face long runtimes when you are determining relationships, because the program tries to determine all links, even if the user does not need all object types.

For example, a user may be interested in the logistical chain of business processes, but not for all details in accounting. Here it would make sense to simply hide the unwanted object types. The tool for configuring a more selective display is the personalization. Depending on whether the settings are valid for individual users or for a role, you can call up the personalization tool either via User Maintenance (transaction SU01) or Role Maintenance (transaction PFCG). Settings made for a role can be automatically copied to all users that are assigned to this role. In the role “Document Relationship Browser” the selection of object types is set so that all object types are displayed.

When you hide object types, keep in mind that the documents in question are not only removed from the display, but also that they cannot be used for determining other relationships. This means that not only those objects that were explicitly hidden from the display are excluded from the display, but also the objects that are dependent on the hidden objects.

5.5.3.4 Choosing Fields in DRB

In the DRB navigation tree the default display only includes the type and description of the object. You can expand this display by adding other relevant fields. In addition to the technical fields of the object key and the object type, particularly two fields are important here:

The field Logical System shows in which logical system the data originates. This is relevant when you are looking at cross-system processes or business events. In the context of data archiving the field Origin is particularly relevant. It shows whether a displayed business object is in the database or in the archive.

Similar to the procedure for entry lists, here you can use layouts to make your field selection, save user-specific layouts, and create defaults.

5.5.3.5 Improved Performance in the DRB Tree

When displaying the document relationships in the DRB tree, DRB always determines the links one step ahead of the step that is currently being displayed. The expand symbol only appears for those nodes that have a subnode. This process may negatively affect performance when you call up DRB and when you try to expand documents in the DRB tree. This can be prevented by choosing the Optimum Performance option in the DRB personalization function under Settings for DRB Tree. If this is set DRB only determines the nodes that are to be displayed. You can also choose the Optimum Display option if performance is not an issue.

Beachten Sie auch die SAP-Hinweise 558462 und 497820 zu weiteren Laufzeitverbesserungen im DRB. For further measures to improve the runtime of DRB, see SAP Note 558462 and 497820.
Appendix

A. Glossary

The terms included in this glossary are important for understanding the archiving concept introduced in this handbook. For other terms related to data archiving, see the SAP Terminology Database.

- **Storage system**
  A storage or archive system connected to the SAP System via a certified interface. It is used to store the archive files that are created during the write phase. A storage system can contain a variety of different storage media, but in general it is based on optical storage media such as CD ROMs and WORMs.

- **ADK, see Archive Development Kit**

- **Archive**
  The total of data saved in archive files.

- **Archive administration**
  A function that is called using transaction SARA. It is the central starting point for most user activities in data archiving, such as the planning of write and delete jobs, building and deleting indexes, and storing and retrieving archive files.

- **Archive file**
  A file that is created in the file system of the SAP system by the write program. It contains the archived data. The maximum size of an archive file can be determined by the user. It can have one or several data objects and belongs to exactly one archiving session.

- **Archive Development Kit**
  Abbreviation ADK. The technical framework and basis for the SAP data archiving concept. ADK is a software layer between the SAP applications and the archive. It is the runtime and administration environment for most of the functions of SAP data archiving. ADK also provides a programming interface (ADK API) for the development of archiving programs by SAP or customers.

- **ArchiveLink**
  The interface used to control communication between an SAP application and an external storage system. It can be used to store data and documents created within SAP applications, and also to access this data.

- **Archive Information Structure**
  The central element of the archive information system. It is a kind of index which is created based on a field catalog (see Field catalog) and is used to find archive data in the archive.

- **Archive Information System**
  Abbreviation AS. A generic tool used to conduct searches in archives. It is integrated into the data archiving process. Searching and displaying the data is based on archive information structures, which are defined by the user and can be filled with data from the archive.

- **Archiving session**
  An archiving unit which is made up of a write and delete phase and an optional storage phase. In addition to the actual archiving process the data set written to the archive during the write phase is also called the archiving session. This session can be viewed as a whole in archive management under a unique ID, the archive session number.

- **Archiving object**
  A logical object that encompasses all the application data linked through a business process that must be written to archive files and then deleted from the database. In addition it comprises the corresponding archiving programs and the Customizing.

- **Archiving programs**
  The general term for all programs used during archiving, such as write, delete, read and reload programs.
• **Archiving class**
  A mechanism used to archive certain data shared by different business objects, together with data objects from applications. Examples of archiving classes are SAPscript texts, change documents and classification data.

• **Archive management**
  The part of Archive Administration in which archiving sessions and archive files for a specific archiving object are displayed and managed. It can be reached via Archive Administration (transaction SARA).

• **Business object**
  The representation of a central business object from a real life scenario, such as an order or an invoice. From a technical viewpoint a business object is an instance of a business object type with concrete values. Business objects are managed in the Business Object Repository (BOR). See also **Data object**.

• **Code page**
  Vendor- and hardware-specific coding that assigns a hexadecimal value to each character in a character set.

• **Data object**
  A logical data processing unit used in ADK. It contains application data that is linked through a business process. During data archiving it is written as a whole to the archive file by the write program. It is similar to a business object in an application context (see **Business object**). The archiving object determines from which database tables data records are incorporated into the data object (see also **Archiving object**).

• **Direct access**
  Also called Single document access. A read access to individual data objects in the archive. The pointer is positioned at the beginning of the data object within the archive file. Here only the data object specified during the selection is read (for example Invoice). This method uses an index, which can be generated with, for example, the Archive Information System.

• **Document storage**
  The electronic storage and management of documents, such as original documents, outgoing documents, and print lists, on an external storage system. See also **Optical archiving**.

• **Print list**
  The result of an application program, presented in list form. The print list can be printed on paper and/or stored in a storage system using ArchiveLink.

• **Field catalog**
  A collection of fields that can be used to create or maintain an archive information structure.

• **HSM system**
  The abbreviation used for Hierarchical Storage Management System. This storage solution automatically distributes data according to individual rules (for example frequency of data accesses) across a hierarchy of different storage media (such as hard disc, MO disc, magnetic tapes). To the accessing system the HSM system is a file system that stores files under a logically unchangeable file path.

• **Infostructure, see Archive Information Structure**

• **Jukebox**
  An automated storage unit with different storage media. It is composed of disc drives, storage compartments, and a robot mechanism that automatically changes the optical discs. It facilitates access to comprehensive data archives, without the need for intervention by an operator.

• **Knowledge provider**
  Abbreviation KPro. A central SAP Basis service of the SAP Web Application Server, used for the storage and management of any documents or document-like objects.

• **KPro, see Knowledge provider**

• **Read program**
A program that reads and evaluates archived data or outputs this data in list form.

- **Delete program**
  A program that first reads the data previously written to the archive by the write program and deletes the corresponding data from the database.

- **Delete job**
  The execution of the delete program in the background. The delete job can be executed before or after the archive file has been stored in the storage system, depending on the Customizing settings.

- **Delete phase**
  The part of the archiving session during which data is deleted from the database. The delete phase of an archiving session begins with the start of the delete program for the first archive file. It ends when all files that belong to the archiving session have the status *Deletion Complete*.

- **Meta data**
  The information stored in archive files, which is used to achieve a platform-independent storage and interpretation of the stored data. Examples: Schemas of the database tables, data type and length of a column, number format, code page.

- **Postprocessing program**
  A program that can be executed after the delete phase to perform further application-specific procedures on the data in the database (for example the updating of statistics).

- **Offset**
  A value in relative addressing that specifies how far a certain element or a specific position is located away from the starting point. The starting position of a data object in an archive file is specified by an offset.

- **Optical archiving**
  A widely used, although unfitting, term for document storage. The storage system used in this process is based on optical media, such as CDs and WORM, which is why the expression refers to "optical" archiving.

- **Residence time**
  The amount of time that has to have passed, before application data can be archived. The basis for calculating the residence time can be the entry date, the posting period, the goods issue date, depending on the application. It is usually expressed in number of days.

- **Reload program**
  A program that reads archived data from the archive and reloads it into the database.

- **SARA, see Archive Administration**

- **Write program**
  A program that selects the data to be archived in the database and writes them to one or more archive files.

- **Write job**
  The execution of the write program in the background.

- **Write phase**
  The part of the archiving session during which data is written from the database to an archive file. The write phase of an archiving session begins and ends with the execution of the write program.

- **Sequential reading**
  A read access to archived data. The pointer is positioned directly at the beginning of the archive file and is moved on sequentially. The Archive Development Kit transfers these data objects to the read program, which compares the data with the selection criteria entered by the user and evaluates or outputs the matching data. The read process is completed as soon as the end of the file has been reached. When
several archive files or archiving sessions are evaluated the read process ends when the last file end has been reached.

- **Management data**
  The additional information about archive files and archiving sessions. Management data is stored in the database. Examples: Number and size of data objects, archiving status, logical file path, physical file name, all archive files that belong on one archiving session, etc.

- **Preprocessing program**
  A program that can be executed before the write phase, to check the data for archivable and to prepare the data for archiving, by, for example, setting a deletion indicator.

## B. Additional Information and Services

### SAP Service Marketplace

For a wide range of additional SAP data archiving information see the SAP Service Marketplace (http://service.sap.com) under /DATA-ARCHIVING and /ADK.

### SAP Library

The following table contains an overview of the archiving documentation in the various solutions or components of the mySAP Business Suite. You can find this documentation either via the SAP Library or the SAP Help Portal (http://help.sap.com).

<table>
<thead>
<tr>
<th>Component</th>
<th>Path in the SAP Library</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP NetWeaver</td>
<td><strong>SAP Web Application Server</strong> → <strong>Cross-Application Functions</strong> → <strong>Archiving Application Data</strong></td>
<td>Introduction to data archiving, and archiving of data in SAP Web AS</td>
</tr>
<tr>
<td></td>
<td><strong>ABAP Workbench (BC-DWB)</strong> → <strong>BC Extended Applications Function Library</strong> → <strong>Archive Development Kit</strong></td>
<td>Development of archiving objects and programs</td>
</tr>
<tr>
<td></td>
<td><strong>SAP Business Information Warehouse</strong> → <strong>System Administration Tasks</strong> → <strong>Data Archiving</strong></td>
<td>Archiving data in infocubes and ODS objects in the SAP Business Information Warehouse</td>
</tr>
<tr>
<td>SAP R/3 (4.6C)</td>
<td><strong>Cross-Application Components</strong> → <strong>Archiving of Application Data (CA-ARC)</strong></td>
<td>Data archiving specific to the different applications in SAP R/3</td>
</tr>
<tr>
<td></td>
<td><strong>Basis Components</strong> → <strong>ABAP Workbench (BC-DWB) → BC Extended Function Library → Data Archiving - ADK</strong></td>
<td>Development of archiving objects and programs</td>
</tr>
<tr>
<td>SAP R/3 Enterprise (4.70)</td>
<td><strong>SAP R/3 Enterprise Application Components</strong> → <strong>Scenarios in Applications</strong> → <strong>Data Archiving(CA-ARC)</strong></td>
<td>Data archiving specific to the different applications in SAP R/3</td>
</tr>
<tr>
<td>SAP Customer Relationship Management (4.0)</td>
<td><strong>Application Services</strong> → <strong>Application Engines</strong> → <strong>Data Archiving</strong></td>
<td>Data archiving in CRM Enterprise</td>
</tr>
</tbody>
</table>

**Table 3: Archiving Documentation in the SAP Library**

### Archiving Your SAP Data

This book provides you with a comprehensive overview of the data archiving technology and how to implement data archiving projects, including all relevant processes. The information is mainly focused on
SAP R/3 Enterprise, but it also introduces newer archiving concepts, such as for CRM server or the SAP BW. The book was written by a team of SAP archiving experts and provides an extensive array of details and practical recommendations taken from real life data archiving scenarios. It is also available in German and Japanese.

Further information:

Archiving Your SAP Data
Helmut Stefani
SAP Press
334 Pages, 2003
ISBN 1-59229-008-6
60.00 USD

Training Courses

- BIT660 – Data Archiving
  Three-day introductory course to the technology and concepts of SAP data archiving. Data archiving is explained using examples from the most important SAP R/3 components.

- BIT670 – Data Archiving (Programming)
  This two-day course explains how to develop read and analysis programs for archived data.

- BIT614 – Document Management at SAP: An Overview
  Overview of the different document management options in an SAP system. The focus of this two-day course is the storage and management of documents using different SAP components such as SAP Content Server, ArchiveLink, Records Management, Knowledge Warehouse, etc.

- BIT615 – Document Storage using ArchiveLink
  Introduction to document storage using ArchiveLink. This two-day course discusses the storage of original documents in a storage system and their relationship with SAP business objects.

- WDE680 – Data Retention Tool
  Introduction to the Data Retention Tool (DART). DART is a tool used in the context of GDPdU to help provide data extracts to financial authorities during audits.

Consulting and Services

- Consulting
  When you carry out an archiving project it may be necessary to call in an external consultant to help you. SAP consulting can offer you the services of competent consultants with comprehensive know-how and extensive experience in the data archiving field. To contact SAP consulting go to the SAP Service Marketplace and choose the menu option Consulting.

- SAP Data Archiving Optimization
  This SAP service, available in the SAP Service Marketplace at /DAO, helps you tune your archiving process to best fit the technical characteristics and business process in your SAP solution, so that you can achieve the most benefits from data archiving.

- SAP Data Management
  This SAP service helps you determine how you can save more disc space in your database. Based on a comprehensive database analysis, you can optimize the use of resources in your database through measures such as data prevention, aggregation, deletion and archiving. This service is available in the SAP Service Marketplace under /DMA.