infotel

Arcsys

Prerequisites Guide

25.3.1.STS
December 9, 2025

Version: 25.3.1.STS

Publication date of the document: 2025-12-09

This document is intended for anyone who requires more information on the Arcsys product.

This document is written for all owners of a valid license.

The reader agrees to respect the confidential nature of this document.

Contact details:

France	***Germany***	***USA***
INFOTEL SA *Le Valmy, 6/8/18 Avenue Léon Gaumont* *F-75020 Paris* *France*	*Insoft Infotel Software GmbH* *Sternstr. 9-11* *D-40479 Düsseldorf* *Deutschland*	*INFOTEL Corporation* *PO Box 47517 Florida* *33743 St Petersburg* *United States*
+33 (0)1 48 97 38 38	*+49 (0) 211 44 03 16-6*	*800 543 1982 – Toll-free telephone (US only)* *+1 727 343 5958*
https://techsupport.infotel.com `<software@infotel.com>`	https://techsupport.insoft-software.com `<software@insoft-infotel.com>`	https://techsupport.infotel-consulting.co.uk `<software@infotel.com>`

Table of Contents

List of Tables

Preface

This document describes all the hardware and software requirements for the Arcsys product (core and related associated modules).

1. Reference Documents

1.1. Concepts

Arcsys Presentation Manual: **Arcsys-presentation-25.3.1.STS-en.pdf**

Arcsys Functional Description Manual: **Arcsys-functional-description-25.3.1.STS-en.pdf**

1.2. Installing and Updating

Arcsys Prerequisites Manual: **Arcsys-requirements-25.3.1.STS-en.pdf**

Arcsys Installation Manual: **Arcsys-installation-25.3.1.STS-en.pdf**

1.3. Operations

Arcsys Administration Manual: **Arcsys-administration-25.3.1.STS-en.pdf**

Arcsys Errors Manual: **Arcsys-error-25.3.1.STS-en.pdf**

1.4. GUI

Arcsys Web Interface User Manual: **Arcsys-web-25.3.1.STS-en.pdf**

Interface Guide: **Arcsys-web-end-user-25.3.1.STS-en.pdf**

1.5. Development

Arcsys API Manual: **Arcsys-api-25.3.1.STS-en.pdf**

1.6. Option guides

ArcHP Option Guide: **Arcsys-option-archp-25.3.1.STS-en.pdf**

ArcREF Option Guide: **Arcsys-option-arcref-25.3.1.STS-en.pdf**

1.7. Optional modules

BatchReporting: **BatchReporting-UserGuide-25.3.1.STS-en.pdf**

ClassAssigner: **ClassAssigner-UserGuide-25.3.1.STS-en.pdf**

MetadataReplacement: **MetadataReplacement-UserGuide-25.3.1.STS-en.pdf**

StartRetentionDateAssigner: **StartRetentionDateAssigner-UserGuide-25.3.1.STS-en.pdf**

2. Symbols and Meanings

Note

Identifies information of particular interest

Important

Identifies important information

3. Definitions and Abbreviations

See the Glossary

1. Symbols and Meanings

Arcsys is designed according to an **open** principle, without a strict adherence to technologies or platforms.

In Arcsys, a distinction is made between:

- the **Arcsys core**, which is a minimal set of mandatory Arcsys components (Arcsys RMI, TCP/IP and SOAP API, Arcsys Transfer Server, Arcsys Engine,Arcsys Application Agent, Arcsys Web Agent, Arcsys Transfer Service, Arcsys Auto-Archive Agent). The Arcsys core can be made up of **options** (for example, AFP management, compression, etc.).
- **the Arcsys optional modules associated with the core**. These modules are: Arcsys REST API, Arcsys standard Clients, ArcsysFsComparator File systems comparator, ArcsysBatchs batch module, ArcFF format control module and CopyRequestManager. They are included in the standard Arcsys installation package and do not require an additional license.
- **the connectors and other optional modules**: for example ArcIP, ArcEP, etc.

This manual describes the general compatibility rules of the Arcsys core (with its options) and its associated optional modules. The options can introduce restrictions with respect to the general rules. For the sake of simplicity, these modules will henceforth be called "**Arcsys** modules".

There are, generally speaking, two types of nodes:

- A **complete Arcsys node** contains all the Arcsys modules;
- A **client Arcsys node** is used to access client data but does not contain the server modules for storage. It contains at least one Arcsys Application Agent, one Arcsys Transfer Service and one Arcsys Auto-Archive Agent.

Subsequently, the expression " is compatible with " means that Arcsys is compatible with this environment, which is then qualified and supported by Infotel.

New environments are regularly qualified, which is why it is highly recommended you contact the Infotel tech support if a target environment does not appear in this document.

2. Requirements for the Arcsys Modules

2.1. Introduction

This section describes the required hardware and software valid for the Arcsys modules.

2.2. System Resources

Arcsys has a number of requirements at system resource level: RAM, CPU and disk space.

Important

These necessary resources vary significantly depending on how Arcsys is used (data volume, number of simultaneous accesses, archive breakdown, amount of metadata, etc.). This document provides the basic rules, but we recommended you to contact Infotel for guidelines on initial dimensioning. The following paragraphs outline the elements that Infotel will need to provide adapted recommendations.

2.2.1. RAM

2.2.1.1. Variable Factors

Memory size used by Arcsys processes is not set and depends on a wide range of factors. For example, these include:

- The activity taking place on the product, irrespective of its type (archiving, retrieval, viewing, migration,etc.);
- The parallelism desired in the various modules (number of threads at Arcsys Transfer Server level, number of requests managed simultaneously by the Arcsys Engine);
- The operating system characteristics and the total physical memory available;
- The use of optional modules, connectors or specific clients using API functions, in particular, reporting methods sending record or metadata lists;
- The buffer size configured at Arcsys Transfer Server level;
- The number of simultaneous access authorized on each file system in ArcMover Disk, according to the number of simultaneous operations.

Given the number of variable factors, creating dimensioning graphs is not straightforward.

2.2.1.2. Maximum Authorized Memory

At the level of the maximum memory that can be used, the behavior of core Arcsys modules differs according to their nature:

- Java modules: Arcsys RMI, TCP/IP and SOAP API, Arcsys Engine, Arcsys Application Agent, optional Arcsys modules associated with the core. In this case, the maximum memory allocate for these modules is that determined by default by the JVM. Generally speaking, on a server with more than 8 GB of memory, it is set to a quarter of the memory used within the limit of 8 GB. If "OutOfMemory" type errors are noted in the log for these modules, the default value should be increased using the `Xmx` option at module startup script level (or in the command line associated with startup in service mode in the case of Windows).
- JEE application: the Arcsys Web Agent. The maximum memory is generally configured at startup script level of the application server used. See the manual for this application server for more details on this configuration.
- Native binaries: Arcsys Transfer Server, Arcsys Transfer Service. The maximum memory is that allocated by the system. There are no parameters at Arcsys level.

2.2.1.3. Current Recommendations

For example, a **production** configuration corresponding to a **complete Arcsys node** has generally between **8 and 32 GB** of memory. A **production** configuration corresponding to a **client Arcsys node** can function with 4 GB.

2.2.2. CPU

The guidelines depend on a number of factors. Arcsys ensures scalability with respect to the number of processors and cores.

2.2.2.1. Current Recommendations

For example, a **production** configuration corresponding to a complete Arcsys node can be based on a last-generation Intel Xeon with 8 to 32 cores.

2.2.3. Network

An Ethernet network adaptor of 1 Gb is the minimum required in production mode.

2.2.3.1. Current Recommendations

For optimal performance within the framework of archiving of large data volumes, an Ethernet of 10 Gb network is recommended.

2.2.3.2. Internet access for ArcVERIF

Internet access may be necessary for certificate validation, particularly when using methods like CRL (Certificate Revocation Lists) and OCSP (Online Certificate Status Protocol). However, it's important to note that internet access might be required for other aspects of certificate validation beyond CRL and OCSP. For instance, certificate chain verification or retrieving revocation data from alternative online sources may also necessitate internet connectivity. In some environments, such as those where CRLs are stored and updated locally or where a local revocation infrastructure is employed, internet access may not be necessary for certificate validation. For further details on configuring these settings, please refer to the module's administration manual (Arcsys Administration Manual.

2.2.4. Disk Space

In addition to the space to be reserved for record storage, the disk space required for all Arcsys modules is 1GB. Disk space must also be reserved for the current application operation. This space is detailed below.

2.2.4.1. Exchange and Temporary Storage Directories

This disk space varies according to the following choices:

- Size desired for the Arcsys Transfer Server cache;
- If online retention is desired, which is generally recommended when using the ArcMover Tape Option, the retention period desired in the online directory of the Arcsys Transfer Server;
- Maximum size of archiving, retrieval, synchronous retrieval and other operation that could occur simultaneously;
- Use of compression (ArcPAK Option) that requires disk space at the time of decompression;
- Use of regroup by Spooler option for zones with ArcMover Tape Option.
- For ArcVERIF with Signature validator module, a temporary directory must be defined for the web service operations (see the module configuration in the Arcsys

Administration Manual). Approximately 50 Mb are needed to store certificates (XML trust lists) in the temporary directory. Additionally, if the Signature Validator WS is not installed on the same server as ArcFF format control module, ensure sufficient disk space corresponding to the size of the files to be validated simultaneously.

For more details, please refer to the Arcsys Administration Manual .

2.2.4.2. Logs and Traces

Arcsys generates logs and traces in proportion to the activity. You must provide enough disk space for one day of logs, as these logs are auto-archived daily. In **production** mode, log files usually represent 10 to 15% of the volume archived.

Note

Log and trace files can be rewritten by symbolic links (with Unix) to files stored elsewhere than in the installation directory of modules.

2.3. Database

2.3.1. Database Compatibility

Arcsys is compatible with the following databases:

- MariaDB version 10.11, in UTF-8 mode for a first-time Arcsys installation.
- Oracle Database version 19c; (Oracle Database version 21c compatibility has not been validated at the time of release of this document).

Note

The Oracle Database RAC option is supported.

- Microsoft SQL Server version 2019.
- PostgreSQL 15 and 15.x minor releases are recommended (but PostgreSQL 14 and 14.x minor releases are also supported)

2.3.2. Access to the Database

The hosts of the modules that access the Arcsys Database (Arcsys RMI, TCP/IP and SOAP API, Arcsys Web Agent, Arcsys Engine, Arcsys REST API) must have sufficient permissions for the relational database.

Important

This point is essential. If not applied, connection problems may arise.

2.3.3. Dimensioning the Database

Dimensioning the disk space reserved for storage in the Arcsys Database depends on a number of factors (such as type of database, configuration, number of records and documents, amount of metadata). However, it does not depend on the volume of records. When initially dimensioning the database, please contact Infotel support for guidelines adapted according to your use.

2.3.4. Database Encoding

It is recommended to configure the database in UTF-8, so that Arcsys can store information (metadata, labels, file names, etc.) in UTF-8 format.

- With Oracle, we recommend you to have an instance configured with "`Database character set`" set to "AL32UTF8".

  The database must be case-sensitive; this ensures that character strings are case sensitive. The default Oracle configuration is `NLS_COMP=BINARY`.

- MariaDB must be configured in UTF-8. There are two ways to perform this configuration:

  - By configuring MariaDB in UTF-8 by default. This is done by modifying the MariaDB configuration file. In this case, if there are no specification during Arcsys database creation, UTF-8 encoding applies by default.

    To configure MariaDB in UTF-8 by default, specify the following sections in its `my.cnf` configuration file:

    ```
    [client]
    default-character-set=utf8

    [mysqld]
    collation-server = utf8_bin
    init-connect='SET NAMES utf8'
    character-set-server = utf8

    [mysql]
    default-character-set=utf8
    ```

  - Or by specifying the configuration only for the Arcsys database. This is done while creating the latest in the `create database` statement.

    For example:

```
create database archiving_product character set utf8 collate utf8_bin;
```

Important

On a first-time Arcsys installation, a MariaDB database must mandatorily be configured in UTF-8. On an upgrade from Arcsys prior to 5.0 version, if MariaDB is configured in another encoding, this encoding must not be changed without explicit instructions from the Infotel support.

- For Microsoft SQL Server: UTF-8 encoding is introduced in Microsoft SQL Server 2019.
- For PostgreSQL: UTF-8 encoding is mandatory.

2.3.5. Database Preparation

Before installing Arcsys, you must prepare the target database. This includes creating the required user and configuring the database according to the system you are using.

The following subsections provide guidelines and examples for each supported database system.

2.3.5.1. Oracle Database

For Oracle Database, create at least two tablespaces before running the installer: one for indexes and one for data. The installer will prompt you to provide the names of these tablespaces during installation.

Creating a User and Tablespaces

Use the following commands to create a user and tablespaces in Oracle Database using the pluggable database (PDB) `ARCHIVING_PRODUCT_PDB`:

```
ALTER SESSION SET CONTAINER = ARCHIVING_PRODUCT_PDB;
CREATE TABLESPACE ARCHIVING_PRODUCT_DATA DATAFILE '/opt/Infotel/ArchivingProduct/
fororacle/ARCHIVING_PRODUCT_DATA.dbf' SIZE 2G REUSE AUTOEXTEND ON BLOCKSIZE 8192 EXTENT
 MANAGEMENT LOCAL AUTOALLOCATE;
CREATE TABLESPACE ARCHIVING_PRODUCT_INDEX DATAFILE '/opt/Infotel/ArchivingProduct/
fororacle/ARCHIVING_PRODUCT_INDEX.dbf' SIZE 2G REUSE AUTOEXTEND ON BLOCKSIZE 8192 EXTENT
 MANAGEMENT LOCAL AUTOALLOCATE;
CREATE TEMPORARY TABLESPACE ARCHIVING_PRODUCT_DATA_tmp TEMPFILE '/opt/Infotel/
ArchivingProduct/fororacle/ARCHIVING_PRODUCT_DATA_tmp.dbf' SIZE 500M AUTOEXTEND ON EXTENT
 MANAGEMENT LOCAL UNIFORM SIZE 512K;
CREATE USER ARCHIVING_PRODUCT IDENTIFIED BY password CONTAINER = CURRENT DEFAULT TABLESPACE
 ARCHIVING_PRODUCT_DATA TEMPORARY TABLESPACE ARCHIVING_PRODUCT_DATA_tmp QUOTA UNLIMITED ON
 ARCHIVING_PRODUCT_DATA QUOTA UNLIMITED ON ARCHIVING_PRODUCT_INDEX;
```

```
-- Roles
GRANT CREATE DATABASE LINK TO ARCHIVING_PRODUCT;
GRANT CREATE CLUSTER TO ARCHIVING_PRODUCT;
GRANT CREATE PROCEDURE TO ARCHIVING_PRODUCT;
GRANT CREATE SEQUENCE TO ARCHIVING_PRODUCT;
GRANT CREATE SESSION TO ARCHIVING_PRODUCT;
GRANT CREATE SYNONYM TO ARCHIVING_PRODUCT;
GRANT CREATE TABLE TO ARCHIVING_PRODUCT;
GRANT CREATE TYPE TO ARCHIVING_PRODUCT;
GRANT CREATE VIEW TO ARCHIVING_PRODUCT;

-- Privileges
GRANT CREATE TRIGGER TO ARCHIVING_PRODUCT;
GRANT SELECT ANY DICTIONARY TO ARCHIVING_PRODUCT;
GRANT EXECUTE ON DBMS_RANDOM TO ARCHIVING_PRODUCT;
```

For more information on PDBs, refer to the official Oracle Database documentation.

Note

After installation, consider moving large, periodically cleanable tables (e.g., `JOB` and `JOB_ACTIONS`) to dedicated tablespaces. This helps optimize disk usage and does not impact the upgrade process. Contact technical support for assistance with space management.

2.3.5.2. MariaDB

To prepare MariaDB, create the database and user before running the installer:

```
CREATE DATABASE archiving_product CHARACTER SET utf8 COLLATE utf8_bin;
GRANT CREATE, DROP, SELECT, INSERT, UPDATE, DELETE, CREATE ROUTINE, ALTER,
 TRIGGER, ALTER ROUTINE, INDEX ON archiving_product.* TO 'archiving_user'@'%'
 IDENTIFIED BY 'password';
GRANT CREATE, DROP, SELECT, INSERT, UPDATE, DELETE, CREATE ROUTINE, ALTER, TRIGGER,
 ALTER ROUTINE, INDEX ON archiving_product.* TO 'archiving_user'@'localhost'
 IDENTIFIED BY 'password';
```

Important

If binary logging is enabled, you may encounter the error: "You do not have the SUPER privilege and binary logging is enabled." To resolve this, add the following line to the MariaDB configuration file `my.cnf`, immediately after the `log-bin=...` line: `log-bin-trust-function-creators=1`.

2.3.5.3. Microsoft SQL Server

Before running the installer, create the target database in Microsoft SQL Server.

The following is an example of how to create the database using SQL Server 2019. Replace `NN` with the appropriate size values based on your requirements:

```
USE master;
CREATE DATABASE ArchivingProductDB ON
(
    NAME = "ArchivingProductDB_dat",
    FILENAME = "C:\Program Files\Microsoft SQL Server\mssql\ArchivingProductDB.mdf",
    SIZE = NN,
    MAXSIZE = NN,
    FILEGROWTH = NN
)
COLLATE Latin1_General_100_CI_AS_SC_UTF8;
```

2.3.5.4. PostgreSQL

To prepare PostgreSQL, create the database and user before running the installer:

```
CREATE USER ARCHIVING_USER WITH LOGIN PASSWORD 'password';
CREATE DATABASE ARCHIVING_PRODUCT
    WITH ENCODING='UTF8'
    OWNER=ARCHIVING_USER
    CONNECTION LIMIT=-1;
GRANT ALL ON SCHEMA PUBLIC TO ARCHIVING_USER;
GRANT ALL ON ALL TABLES IN SCHEMA PUBLIC TO ARCHIVING_USER;
```

Note

Ensure remote access permissions are configured in the `pg_hba.conf` file.

For example:

```
host archiving_product archiving_user 0.0.0.0/0 password
```

2.4. OS

Due to the use of native binaries (Arcsys Transfer Server, Arcsys Transfer Service) or native libraries for Java modules, restrictions are in place for operating systems supported by default. The operating systems currently compatible with Arcsys are as follows:

2.4.1. Operating Systems Supported

Operating System	Minimum OS version	Architecture Supported	Examples
Linux	Version such as glibc >=2.28 / kernel >=4.18	64 bits	Debian 11.0; RHEL/Oracle Linux 8
Windows	Server 2022	64 bits	Windows Server 2022

Table 2.1. Operating Systems Supported

2.4.2. Additional requirements for Linux:

- systemd must be installed on the system where Arcsys is run, even if Arcsys is not run as a systemd service.

2.4.3. Additional requirements for Windows:

- the runtime **Microsoft Visual C++ 2019 Redistributable Package (x64)** is a prerequisite for Arcsys Transfer Server and Arcsys Transfer Service. See https://aka.ms/vs/17/release/vc_redist.x64.exe for example.

2.5. Java Compatibility

Arcsys is a Java-based application. Arcsys requires a JRE **Java 17** 64 bit both for the operation of its modules as well for the operation of the installation and update package. The supported JVMs are *Oracle Java* and *OpenJDK*.

Depending on the editor's distribution policy, the JRE may be exclusively bundled within the JDK package rather than being distributed separately.

Important

Java must be installed in such a way that the `java` command is available in command prompt and is linked to the required Java version. This may require you to modify the `PATH` environment variable and/or settings.

To check the version of java used, enter the `java -version` command in a command prompt and check that the first line displayed gives a version starting with 17. Example: `openjdk version "17.0.9 " 2023-10-17`

Note

The procedure for declaring a JVM and the "default" JVM varies according to the system. For instance, if you use the command **update-alternatives**:

```
# update-alternatives --install "/usr/bin/java" "java" "/opt/jdk17/bin/java" 1
# update-alternatives --set java /opt/jdk17/bin/java
```

2.6. Web Server Compatibility

The Arcsys Web Agent is a WAR application compiled in Java 17 LTS. It requires an application server compatible with Java 17 LTS and `jakarta.servlet` in version 6. Here are some examples of compatible server applications:

- WildFly 27
- Apache Tomcat 10.1
- JBoss EAP 7.4 Update 8, JBoss EAP 8

Important

WildFly follows a unique branching model, and previous releases are not patched. This policy should be considered when choosing your web server and planning your upgrade strategy.

2.7. LDAP Directories and SSO systems

Arcsys is compatible with the LDAPv3 protocol. As such, it is particularly compatible with the following LDAP directories:

- 389 Directory Server 2.x
- Active Directory and Active Directory Lightweight Directory Services.

Arcsys Web Agent and ArcWeb Module are also compatible with an Identity Provider using SAML2 protocol for Single Sign On authentication. Please note that a LDAP directory is still necessary in order to load users authorizations.

For example, KeyCloak 4.8.0 and later is compatible.

2.8. Web Browser

Arcsys Web Agent and ArcWeb Module are compatible with the following browsers:

- Microsoft Edge
- Google Chrome
- Mozilla Firefox

"Current" or "Current -1" versions of Edge, Chrome and Firefox are supported when observing any anomalies.

Important

The client browser must authorize Javascript and cookies.

2.9. Time Server

A time server with an external synchronized clock is required. All Arcsys nodes and the database used by Arcsys must use this time server.

2.10. Security

2.10.1. Ports to Open

2.10.1.1. Arcsys Ports

Module	Port default value	Additional information
Arcsys Engine	25020 (TCP)	Parameter *COMPONENT_API_PORT_NUMBER* in ENGINE.properties
Arcsys Application Agent	25030 (TCP)	Parameter *COMPONENT_API_PORT_NUMBER* in APPAGENT.properties
Arcsys RMI API	25040 (TCP)	Parameter *COMPONENT_API_PORT_NUMBER* in ARCSYS_RMI_API.properties
Arcsys TCP/IP API	25050 (TCP)	Parameter *COMPONENT_API_PORT_NUMBER* in ARCSYS_TCPIP_API.properties
Arcsys SOAP API	25060 (TCP)	Parameter *COMPONENT_API_PORT_NUMBER* in ARCSYS_WS_API.properties
ArcFF format control module	8085 (TCP)	Parameter *JETTY_SERVER_PORT* in arcff.properties
Arcsys Transfer Service	25010 (TCP)	Parameter *component_port_number* in transferService.conf
Arcsys Transfer Server	25010 (TCP)	Parameter *component_port_number* in transferServer.conf
Arcsys REST API	8090 (TCP)	Parameter *HTTP_PORT* in arcsys_rest.properties

Table 2.2. Arcsys Ports

2.10.1.2. Ports for Other Applications

You must open the ports:

- for the database (for example, tcp/1521 for Oracle)
- for the application server
- of the LDAP directory (for example, tcp/636)
- where appropriate, the media manager (Cloud provider).

2.10.2. Antivirus Compatibility

To ensure the proper functioning of the product, it is essential to verify that any antivirus software installed on the system does **not block or intercept HTTPS requests**. Such interference may disrupt components such as *Arcsys REST API* and *ArcFF format control module*, which rely on REST APIs for communication and status monitoring.

In particular, the status check mechanism in these components sends an HTTPS request to determine whether the target component is running. If this request is intercepted, modified, or blocked by the antivirus, the returned response may be incorrect. This can result in a false positive, indicating that the component is up when it is in fact stopped.

Additionally, to avoid performance degradation and potential access conflicts, it is strongly recommended to **exclude from antivirus scanning the shared buffer directories** `stage/serv` and `stage/cli`. These directories are used by *Arcsys Transfer Server*, *Arcsys Transfer Service*, and *Arcsys Application Agent* and are subject to frequent read and write operations. Scanning them may interfere with normal execution and significantly affect system performance.

3. Compatibility Limitations

This section describes the limitations introduced by the Arcsys core options.

3.1. ArcMover Tape Option (archiving on tape)

3.1.1. OS

ArcMover Tape Option archiving is compatible with the following operating systems:

- Linux in its versions supported by Arcsys Core

3.1.2. Tape libraries and Tape Drives

The ArcMover Tape Option is compatible with all tape libraries complying with the SMC3 standard (SCSI Media Changer 3.0).

All drives complying with the SPC4 (SCSI Primary Command 4.0), SSC4 (SCSI Stream Command 4.0) and SBC3 (SCSI Block Command 3.0) standards are compatible.

The drives used by the ArcMover Tape Option must be exclusively allocated to ArcMover Tape.

Arcsys does not support partitioned tapes. An error is raised by the Arcsys Transfer Server when loading a partitioned tape.

Here are a few supported configuration examples:

Oracle StorageTek SL500 Tape libraries (1201 firmware):

- IBM LTO4 ULTRIUM-TD4 drive (C7QH firmware);

Oracle StorageTek SL150 Tape libraries (0407 firmware):

- HP LTO5 Ultrium 5-SCSI drive (Y68S firmware);

Quantum Scalar I40 Tape libraries (180G firmware):

- HP Ultrium 6-SCSI drive (25GZ firmware).

Quantum Scalar I3 Tape libraries (330G firmware):

- IBM ULTRIUM-HH7 drive (Q3A1 firmware);
- IBM ULTRIUM-HH8 drive (P381 firmware);
- IBM ULTRIUM-HH9 drive (R3G3 firmware).

IBM TS3500 Tape libraries (C261 firmware):

- IBM TS1040 LTO4 ULT3580-TD4 drive (C7Q0 firmware);
- IBM TS1130 Jaguar E06 drive.

Important

The ArcMover Tape Option is not compatible with the hardware compression functionality.

3.1.3. Tape labels

Arcsys needs the tape labels to use the standard bar code labels, composed of 6 characters for VOLSER followed by 2 characters for media type identifier.

3.1.4. Spooler

ArcPAK Option for compression and ArcCrypt Option are not supported with the Spooler feature on ArcMover Tape Option.

3.2. ArcPAK Option

3.2.1. Data Compression When Writing to Media

The software data compression when writing to a file system or a tape (configured by the compression level in the storage policy) is available on ArcMover Disk, ArcMover Tape, Cloud and Generic type zones. However, tape spooling is not compatible with data compression.

Important

Lots including ZIP/LZMA native objects (see below) are accepted but will not benefit from being compressed on media.

3.2.2. ZIP or LZMA Native Archiving

3.2.2.1. OS

ZIP or LZMA native archiving is compatible with the following operating systems only:

- Windows in its versions supported by Arcsys Core
- Linux in its versions supported by Arcsys Core

3.2.2.2. Database

ZIP or LZMA native archiving is compatible with an Oracle Database, MariaDB or PostgreSQL database only.

3.2.2.3. Compression method

ZIP, LZMA and LZMA2 compression algorithms are supported.

3.3. ArcAFP Option (AFP file archiving)

3.3.1. OS

ArcAFP Option is compatible with the following operating systems:

- Windows in its versions supported by Arcsys Core
- Linux in its versions supported by Arcsys Core

3.3.2. Database

The ArcAFP Option is compatible with an Oracle Database database only.

3.3.3. Limitations on the PDF generation

The library used for PDF generation does not allow to generate PDF files with more than 32768 pages.

3.4. External Media Manager Compatibility

The Arcsys Transfer Server can archive to various media types. Those medias are accessed by media managers. The Arcsys Transfer Server has an internal media manager called ArcMover (disk medias are accessed with ArcMover Disk, and tape medias with ArcMover Tape). The medias can also be accessed by other software (called by the generic term "external media managers"): Cloud providers (for S3-compatible cloud providers) and Generic media manager.

3.4.1. S3 Cloud compatibility (ArcMOVS3 Option)

The Arcsys Transfer Server is compatible with any Cloud media compatible with the Amazon S3 REST API and supports AWS version 4 signature authentication (for example: Amazon S3, Minio, Ceph Object Gateway version >=10.1 Jewel).

3.4.1.1. OS

Cloud archiving is compatible with the following operating systems only, for the Arcsys Transfer Server:

- Linux in its versions supported by Arcsys Core.

3.4.1.2. External library

The cloud archiving uses the `libcurl` system library. The `libcurl` library must be available in a minimal version 4.5 and must be built with a compatible SSL library.

The list of compatible SSL libraries for `libcurl` can be found on the `libcurl` website: https://curl.haxx.se/docs/ssl-compared.html.

3.4.1.3. Requests type for S3 type of Cloud media

The S3 Cloud media called by the Arcsys Transfer Server must be compatible with the following Amazon S3 REST API requests:

- Existence test of a bucket and an object:
  - `HEAD /Bucket`
  - `HEAD /Bucket/ObjectName`
- Partitioned loading:
  - `POST /Bucket/ObjectName?uploads`
  - `PUT /Bucket/ObjectName?partNumber=PartNumber&uploadId=UploadId`
  - `POST /Bucket/ObjectName?uploadId=UploadId`
  - `DELETE /Bucket/ObjectName?uploadId=UploadId`
  - `GET /Bucket/ObjectName?uploadId=UploadId`
- Object recovery: `GET /Bucket/ObjectName`
- Object deletion:
  - `GET /Bucket/?versioning`
  - `GET /Bucket/?versions&prefix=Prefix`
  - `DELETE /Bucket/ObjectName`

- `DELETE /Bucket/ObjectName?versionId=VersionId`

3.4.1.4. Server-side encryption

Support for Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3) is available on Cloud media manager for type S3. This parameter can only be used to access a cloud-provider supporting x-amz-server-side-encryption header.

Note

Server-side encryption is used to encrypt data at rest on the cloud-provider storage. The data is encrypted by the cloud-provider, after being sent to it.

To encrypt data at Arcsys level, before transferring it to the cloud-provider, use ArcCrypt Option (See Encrypting data at rest with ArcCrypt Option).

3.4.1.5. Object Locking

Object Lock Compatibility

Arcsys is compatible with AWS S3 Object Lock, but it does not enable or manage Object Lock settings. If you want to use Object Lock, you must configure it yourself on the bucket(s) where it is required. Please refer to your cloud storage provider's documentation for setup instructions.

Impact of Object Lock on Arcsys requests

When Object Lock is enabled on a bucket, it has no impact during the archiving process.

During retrieval, Arcsys always returns the latest version of the document, which may not be the archived version. If a file has been altered or compromised by a third party, Arcsys detects that the retrieved data does not correspond to the one archived, and the retrieval request fails with an error. In such cases, it may still be possible to manually recover the original, unaltered data by accessing the cloud storage provider directly. This recovery depends on the Object Lock configuration: the original version must still be within its retention period and protected from deletion.

For deletion operations (whether due to end-of-life retention or storage migration), the behavior depends on the Object Lock configuration:

- If the retention period set by Object Lock has expired, deletion can proceed normally.
- If the retention period is still active, the data cannot be deleted.

Note

The deletion or migration request may still be marked as ended, depending on the value of the `ENFORCE_DELETE_DATA` parameter (see the Arcsys Administration Manual).

Object Lock Configuration Recommendations

To ensure proper data protection, we recommend setting a default retention period on S3 buckets that aligns with your organization's document retention policy. If the Object Lock retention period is shorter than the retention policy set in Arcsys, the document may not be protected for the full duration. If it is longer, deletion of expired documents by Arcsys may fail, depending on the value of the *`ENFORCE_DELETE_DATA`* parameter (see the Arcsys Administration Manual).

3.4.1.6. Replication and versioning

Some Cloud media providers offer replication and history management functionalities (such as for Amazon S3: replication between region and versioning). The use of such tools has no impact on the operation of the Arcsys Transfer Server. Versioning is generally not recommended, as it may lead to additional storage costs billed by the cloud media manager. However, enabling versioning is required if you plan to use Object Lock, in which case it should be configured carefully and aligned with the retention strategy set in Arcsys.

For cross-region replication, we recommend creating two buckets in different regions (each corresponding to a Arcsys zone) and configuring Arcsys to manage replication through storage policies.

For more information on zones and storage policies, see the Arcsys Web Interface User Manual.

Note

When versioning is enabled or suspended on a bucket, the user configured to access the Cloud media provider must have permission to perform the `s3:ListBucketVersions` action on this bucket.

Note

In case a delete marker was placed outside of Arcsys on a document, the document is no longer visible in the bucket, even though it technically still exists as a versioned object. As a result, any attempt to delete the document as part of an end-of-life migration request will fail. The versions of this document will remain on the cloud media.

3.5. Search Engine Compatibility (ArcRFT Option)

3.5.1. GenericSearch version

Arcsys is compatible with GenericSearch version 7 and its minor fixes.

Please check the manuals of GenericSearch for the software and hardware requirements of this module.

3.5.2. Types of documents for full text indexing

Full text indexing is possible only on objects with type **File** or **File of Directory**.

Native objects, such as AFP files or native ZIP files, are not taken into account.

3.6. Encryption

3.6.1. Encrypting data in motion with SSL

When SSL is chosen to encrypt the communications between Arcsys components, the `openssl` library in the 3.x branch is a prerequisite that must be installed on the system. The library is dynamically loaded from a path set in the parameters in the Arcsys Transfer Server configuration file (See Arcsys Administration Manual for more details).

3.6.2. Encrypting data at rest with ArcCrypt Option

This option encrypts data with a data encryption key, which is in turn encrypted with a Master Key. The Master Key is stored in a keystore external to Arcsys.

The keystore may be either:

- a centralized keystore accessed through a HSM using the PKCS #11 API. Supported algorithms are RSA-2048 and AES-256 (RSA-2048 is strongly recommended);
- a local keystore file that follows the PKCS #12 file format for storing cryptography objects. In this case, the RSA-2048 algorithm is the only supported algorithm.

Important

The `openssl` library with a version 3.x is a prerequisite that must be installed on the system, as the Arcsys Transfer Server loads the most recent version found on the system.

Note

Since the encryption is performed using `openssl` in AES256_CBC_PKCS5, it is advised, for best performances, to use a processor having the Intel AES-NI instruction set (on Linux, that can be checked by using the `grep -o aes /proc/cpuinfo` command).

Important

ArcCrypt Option is not supported with the Spooler feature on ArcMover Tape Option.

3.7. ArcHP Option

This option operates with the following restrictions:

- For ArcMover Tape: a changer of tape library cannot be shared between Arcsys Transfer Servers (cluster mode is not supported: no sharing of information or tape reservation).
- For ArcMover Disk: a file system can only be shared between a number of Arcsys Transfer Server on Linux. Windows support is planned for a later date.

4. Preparing for LDAP Authentication

4.1. Overview

Arcsys is based on the organization's LDAP directory for user identification. It also uses this directory to retrieve all information required for correct product operations (groups, permissions, etc.).

For this, you must perform the following operations in the directory:

1. Add a specific `arcsysRight` attribute at the group level.
2. Create Arcsys groups, allocate Arcsys rights.
3. Attach users to these groups.

4.2. Adding the `arcsysRight` Attribute

The `arcsysRight` attribute must be added to the group class.

4.2.1. OpenLDAP

Follow this procedure to update the OpenLDAP directory (OpenLDAP as well as 389 Directory Server, Tivoli Directory Server, etc.)

1. Create a new schema (new `arcsys.schema` file) including the new Arcsys attributes. The contents of this new schema must appear below:

```
attributetype ( 1.3.6.1.4.1.21863.1.1.1.1 NAME 'arcsysRight' DESC 'RFC2256: ARCSYS
 Right' EQUALITY caseIgnoreMatch  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128} )}
```

2. Copy the schema created (`arcsys.schema` file) in the schema directory (normally, the `/schema/` directory).
3. Add the new schema as an include in the `slapd.conf` file. Example of a row to be added:

```
include /etc/openldap/schema/arcsys.schema
```

   This row must be the first row in the includes.

4. Modify the general schema (often `core.schema`) in order to add the `arcsysRight` attribute to the group class (often called `groupOfNames`). This attribute goes into the optional section (MAY).
5. Stop and restart the LDAP server.

4.2.2. Active Directory

Follow this procedure to update an Active Directory directory.

You can perform this procedure using the MMC (Microsoft Management Console).

Use the Active Directory schema to add the attributes.

Use the ADSIEDIT utility to enter these new attributes.

1. In the Active Directory schema, create the `arcsysRight` attribute: name `arcsysRight` , identifying the unique object `1.3.6.1.4.1.21863.1.1.1.1`, syntax `Unicode string`, with multiple values: `yes`
2. In the group class, add the `arcsysRight` attribute.
3. Stop and restart the Active Directory server.

With ADSIEDIT, you can now enter the values for the attributes thus created.

4.3. Creating Functional Groups

Create an `arcsysGroup` group in which you will place all groups specific to Arcsys (Example: Administrator, Reader, ArchiveLauncher, etc.).

4.4. Allocating Rights

For each group created in this way, allocate the appropriate rights For example, for an `Administrator` group, allocate the `ADMIN` right.

Note

Refer to the Arcsys Functional Description Manual for more details.

4.5. Updating Users

Allocate the users who are to use Arcsys to one or more previously created groups.

4.6. Information on Implementing the Kerberos Single Sign On

To implement "server" single sign-on in Arcsys (used for LDAP administrator authentication with the Kerberos ticket system external to Arcsys), you must "link" the LDAP directory and Kerberos.

This configuration is completely optional.

4.6.1. Active Directory

4.6.1.1. Description

The scenario is as follows:

1. The Arcsys user connects to the Unix machine.
2. When the user attempts to use the LDAP service (Active Directory), the authentication via login/password is no longer required.

To reach this result, you must perform a series of operations:

1. Configure Kerberos for Active Directory
2. Add users in Active Directory
3. Install PAM
4. Kerberize PAM

4.6.1.2. Configuring the Kerberos Client

In normal **Kerberos** operation mode, Kerberos is retrieved with its own **KDC**. In this case, point the Unix **Kerberos** to the **Active Directory KDC**.

To do this, simply edit the **Kerberos** configuration files on the Unix machine.

Here is an example of the configuration file of the Kerberos`/etc/krb5/krb5.conf` client with the following features:

- The Active Directory domain is INFOTEL.COM
- The Active Directory directory to which you want to point is on the machine lat102.infotel.com.

```
[libdefaults]
ticket_lifetime = 24000
default_realm = INFOTEL.COM
default_tgs_enctypes = des3-hmac-sha1 des-cbc-crc des-cbc-md5 arcfour-hmac-md5 aes256-cts
default_tkt_enctypes = des3-hmac-sha1 des-cbc-crc des-cbc-md5 arcfour-hmac-md5 aes256-cts
permitted_enctypes = des3-hmac-sha1 des-cbc-crc des-cbc-md5 arcfour-hmac-md5 aes256-cts
[realms]
LABO.INFOTEL.COM = {
   kdc =kerberos.labo.infotel.com:88
   admin_server =kerberos.labo.infotel.com:749
   default_domain =labo.infotel.com
}
INFOTEL.COM = {
    kdc = lat102.infotel.com:88
```

```
    admin_server = lat102.infotel.com:749
    default_domain = infotel.com
}
```

4.6.1.3. Adding Users to ActiveDirectory

You must create a user in Active Directory for each user with SSO.

If the users do not exist in the directory, here is an implementation example:

- Go to **Administration tools** on the Windows machine on which the Active Directory is installed.
- Then go to **Active Directory users and computers**.
- Click on the `Users` folder.
- Click **Create a New User in the Current Container**.
- Enter <**NAME**> in the **First Name** and **User Session Opening Name fields**.

Note

In this case, <**NAME**> can be **user**, for example.

- Click **Next**.
- Enter a password for this Arcsys user (for example, **43210**).
- Select **The password never expires** and **The user cannot change the password**.
- Click **Next** then **Finish**.

You can then immediately test the user on the UNIX machine by:

```
kinit user
Password for user@INFOTEL.COM:43210
New ticket is stored in cache file /tmp/krb5cc_0
```

4.6.1.4. Installing PAM

This document does not cover the procedure for installing PAM; information is available at http://www.installationwiki.org/PAM.

4.6.1.5. Kerberizing PAM

One of the **last steps** in this procedure involves **integrating Kerberos** in the **authentication system**.

You want the **Kerberos tickets** to be created for each user **once on identification** without having to launch the `kinit` command manually after being identified.

Kerberos alone does not replace the **current password files (/etc/passwd, /etc/shadow or /etc/group)**. The **kerberized users** must be present both in the **password** and the **Kerberos** files.

This is possible as each user has **two login/password pairs**, one in `/etc/shadow` and one in the **Kerberos** database. This **Linux-PAM** setup is defined so that either the current authentication or authentication by **Kerberos** should succeed so that a user can connect. In this way, both **users** unknown (**systems** users such as root, daemon, bin, sync, sys, etc.) and known to **Kerberos** can connect.

The *system passwords* in `/etc/shadow` will be tried as a priority. If you want the **Kerberos tickets** to be generated, this authentication mode must **fail** for the **kerberized** *users* (otherwise, the system login is passed, there is no Kerberos authentication and therefore a ticket is not issued).

The most usual way to use **a single password** for **kerberized users** (and for this password to be in **Kerberos**) is to replace their **system passwords** in`/etc/shadow` with **"*K*"**, a string that is **invalid** from an authentication point of view and indicates that the "real" password is **stored in Kerberos**.

This **password** can be set either by modifying the `/etc/shadow` file or by directly typing:

```
usermod -p '*K*' USERNAME.
```

Now add the **Kerberos PAM** module (libpam-krb5 library):

```
apt-get install libpam-krb5
```

Last but not least, edit the following four files as shown here:

`/etc/pam.d/common-account`

```
account sufficient pam_unix.so
account sufficient pam_krb5.so
account required pam_deny.so
```

`/etc/pam.d/common-auth`:

```
auth   sufficient      pam_unix.so nullok_secure
auth   sufficient      pam_krb5.so use_first_pass
auth   required          pam_deny.so
```

`/etc/pam.d/common-password`:

```
password  sufficient    pam_unix.so nullok obscure md5
password sufficient    pam_krb5.so use_first_pass
password required        pam_deny.so
```

`/etc/pam.d/common-session`:

```
session required pam_limits.so
session optional pam_krb5.so
session optional pam_unix.so
```

After modifying **PAM configuration**, restart the services to which you want to connect. This is not always necessary but it prevents any **cache** issue.

For more information on **configuring PAM** files, please check: http://www.kernel.org/pub/linux/libs/pam/Linux-PAMhtml/sag-configuration-file.html

4.6.2. OpenLDAP

4.6.2.1. Description

The purpose of this section is to make the following scenario possible for each Unix machine on which an **Arcsys** service using **LDAP** operates:

1. The **Arcsys** user connects to the Unix machine.
2. When the user attempts to use the **LDAP** service, authentication via **login/password** is no longer required.

The user, like the service, has already been "**kerberized**". Tickets are stored for the user and for the service in a secure location in the Unix machine.

To reach this result, you must perform a series of operations:

1. Add the principles
2. Kerberize LDAP
3. Install PAM
4. Kerberize PAM
5. Launch a Java test client

4.6.2.2. Adding Principles in Kerberos

Perform this operation on each machine hosting an Arcsys module.

Assume the following user launches the module:

```
login: archinving_user
passw: 43210
```

1. Start by connecting to the Kerberos administration console as an administrator: **kadmin**

   You can thus display the list of principles as shown here:

```
list_principals

K/M@INFOTEL.COM
admin/admin@INFOTEL.COM
infotel@INFOTEL.COM
kadmin/admin@INFOTEL.COM
kadmin/bos01.infotel.com@INFOTEL.COM
kadmin/changepw@INFOTEL.COM
kadmin/history@INFOTEL.COM
krbtgt/INFOTEL.COM@INFOTEL.COM
ldap/bos01.infotel.com@INFOTEL.COM
oracle@INFOTEL.COM
```

2. Create a principle with the archiving_user name:

```
addprinc -policy user archinving_user
Enter password for principal "archinving_user@INFOTEL.COM":  43210
Re-enter password for principal "archinving_user@INFOTEL.COM":  43210
Principal "archinving_user@INFOTEL.COM"created.
```

4.6.2.3. Kerberizing OpenLDAP

The purpose of this section is to register the LDAP service in Kerberos so that a user can call it via SSO.

1. Start by connecting to the Kerberos administration console as an administrator:

```
kadmin
```

2. Enter the following command to create a **service key** for **LDAP** with a randomly generated password:

```
addprinc -policy service -randkey ldap/<FQDN>;@<YOUR KERBEROS REALM>
```

   Example:

```
addprinc -policy service -randkey ldap/bos01.infotel.com@INFOTEL.COM
```

3. Create the keytab file associated with the LDAP that contains the private key for the service:

```
ktadd -k <keytab-file-location> ldap/<FQDN>
```

   Example:

```
ktadd -k /etc/ldap/kerberos/ldap.keytab ldap/bos01.infotel.com
```

You can place this file in `/etc/ldap/kerberos/`, for example. You must make note of this location for the remainder of the procedure.

4. Open the `/etc/init.d/slapd` file and add the following line so that `slapd` knows where to retrieve the `keytab` file:

```
export KRB5_KTNAME=/etc/ldap/kerberos/ldap.keytab
```

4.6.2.4. Installing and Kerberizing PAM

This procedure is identical to that for the **Active Directory**.

Glossary

Access Zone

An access zone is an independent entity within Arcsys that defines a controlled network area from which resources can be accessed. These entities can then be attached to permissions (at the repository, collection, lot, or class level) to restrict or grant access based on the client's IP address when authenticating to the Arcsys REST API, the Arcsys Web Agent or ArcWeb Module.

API *(Application Programming Interface)*

The APIs provided by Arcsys enable the product holder to fully customize a new application or user interface according to the specific ergonomic needs of their use case. Arcsys exposes several types of APIs:

- REST APIs are the recommended interface. They offer broad coverage of Arcsys's functionalities, including administration, operations, archiving, search, and archive retrieval.
- Legacy APIs based on RMI and SOAP protocols are still available for compatibility purposes but are deprecated and should no longer be used in new developments.

Application Agent

There are two different types of agents at archiving level: application interface agents and user interface agents. An **application agent** can archive all the objects specific to an application (files, RDBMS table records, etc.), whereas a **web agent** performs both administration functions and manual archiving functions initiated by the user.

Archiving By Reference

Archiving by reference is a method in which data remains in its original storage location when added to an archive system, and the system generates references and metadata entries for the files. Eventually, the files are transferred to the archive system's defined storage using the copy and migration mechanism.

Archive Restitution

Archive restitution is the return and transfer of archived documents to their originator, or to a duly appointed person or organization. An Archive Restitution is in Arcsys an Archive Retrieval operation that ends with a Destruction. An Archive restitution operation can only be created through the appropriate operation in the REST API, or by using ArcEP module.
See Also **Archive Retrieval**, **Destruction**.

Archive Retrieval

Archive retrieval is an operation that makes a copy of a record available to a record requester. This term takes precedence over the term *restore*, which has another meaning at archiving level (restore in the sense of handing back the documents to the organization that created them or to its representatives, then destroying them). Archive retrieval can be

complete (misleadingly called a "complete retrieval") or partial (*Partial Archive Retrieval*, misleadingly called a "partial retrieval").
See Also **Archive Restitution**.

Arcsys
ERM published by Infotel. Arcsys refers to both the Arcsys Core product and all of its connectors and options.

Arcsys Connector
An Arcsys connector is an operational module generally requiring an additional license used to interface with an external software package (ECM, ERP, Mail) for archiving and/or archive retrieval to and from Arcsys.

Arcsys Core
The Arcsys Core represents all "essential" Arcsys modules, which are: Arcsys Database, the Arcsys RMI, TCP/IP and SOAP API, the Arcsys REST API, the Arcsys Transfer Server, the Arcsys Transfer Service, the Arcsys Engine, the Arcsys Web Agent, the Arcsys Application Agent, the Arcsys Auto-Archive Agent, the ArcFF format control module, the CopyRequestManager, the Arcsys standard Clients, the ArcsysFsComparator File systems comparator, the ArcProofFolder Proof Folder module and the ArcsysBatchs batch module.
See Also **Arcsys**.

Arcsys Engine
Central archiving platform on which synchronous and asynchronous archiving, indexing and retrieval processes operate. The engine can spread threads over multiple processors. This guarantees dialogue and traceability between the agents that are associated to it.

Arcsys Option
Arcsys options are added to the Arcsys Core for additional functionalities. They do not necessarily require an additional architectural module. They may be subject to a separate license. The main options are:

- ArcAFP Option (AFP format management)
- ArcMover Tape Option (media manager managing file systems and tape libraries)
- ArcIP (record ingestion)
- ArcEP (record extractor)
- ArcPAK Option (record compression on ArcMover and native ingestion of compressed files)
- ArcRFT Option (full text search)
- ArcSIGN Option (internal digital signature generation) and ArcVERIF (external digital signature verification)

- ArcCrypt Option (encryption of data at rest)
- ArcCFN (digital vault)
- ArcREF Option (record ingestion by reference)
- ArcMOVS3 Option (media manager allowing to archive and retrieve data on any Cloud media compatible with the Amazon S3 REST API)

Attestation policy

An attestation policy allows to define which type of attestation can be generated as well as a set of parameters concerning their generation.

Classification Scheme

A classification scheme in archiving and digital preservation refers to an organized framework for categorizing records and archival materials based on a hierarchical structure. It facilitates systematic retrieval, management, and preservation of information. In the context of Arcsys, the classification scheme is defined as the structural entity that contains a hierarchy of classes. These classes are used for organizing archives and records and for implementing specific archival policies such as retention schedules and format management. Within Arcsys, a classification scheme is linked to a specific repository, providing an organizational backbone for multiple collections. It also serves as a navigational tool for end users, enabling them to explore archives through the hierarchical structure of classes, alongside navigation by repository and collection.

Collection

Set of rules that a record must comply with. The collection is defined via the Web agent or Arcsys API, and comprises information contained in the relational database tables. A collection always refers to two rules: one concerning the **storage policy** and one relating to the **indexing mask**. A collection is assigned to the record on the initial record request.
See Also **Storage policy**, **Indexing mask**.

Deletion

MOREQ2010 provides the following definition for this concept: the act of deleting data from the relational database so that no trace remains. Generally speaking, an entity can only be deleted if is not used in a stored record. Otherwise, it can only be destroyed and not deleted, thus leaving a residual entity.
See Also **Destruction**.

Destruction

Irreversible action that deletes the documents by applying disposal criteria. It can be associated with the retention of residual information in the relational database.

Disposal

Outcome of archived documents when the retention period ends, i.e. generally, destruction or transfer.
See Also **Destruction**, **Transfer**.

Disposal due date (or retention end date)
Scheduled end of retention date.

Disposal Hold
Arcsys can be used to place a "disposal hold" on one or more lots archived in the application. This prevents certain sensitive operations, such as transitioning the lots to end-of-life status or migrating them to a different storage medium. All other operations remain authorized. The disposal hold guarantees that no irreversible change affecting the archival integrity or status of the lot can occur while the hold is active.

Electronic Attestation
Document produced to attest that an action or an electronic transaction has occurred.

Envelope
Arcsys groups documents stored in the system in envelopes, either created by Arcsys during the archiving process (in this case, files in TAR format), or created prior to Arcsys processing by the user or third-party processes (*native envelopes* in AFP or ZIP format, for example). The representation of an envelope in the Arcsys Database is called a **logical envelope**. Its technical implementation is also called *MoverReference*. Last but not least, the representation of information of where the envelope is physically stored in the optional ArcMover module is called *MoverMedia*.

Event
In Arcsys, a retention schedule can associate the start of record retention with an event with a known or unknown date. This event, created in an Arcsys repository, can thus be attached to a number of different retention schedules.
See Also **Retention schedule**.

Feature preview
A Preview status on a feature enables early access to non-production features, allowing users to explore and provide feedback for improvement.

Features in Preview status should not be used in production environment, as they are not fully implemented yet.

Fixity
The quality of a document that has not been subject to intentional or accidental destruction, alteration or modification.

Format policy
A format policy is used to define a set of rules concerning format checks for a given file type. These rules are used to specify the action that will be performed, the expected results of these actions, as well as the error cases, triggering archiving failure.

Hash value
Also called an "integrity certificate" in cryptography, the hash value is the digest of a message which guarantees a practically unique result that is impossible to reverse calculate. The most commonly used algorithms are MD5 (128 bits), SHA-1 (160 bits), SHA256 (256

bits) and SHA512 (512 bits). Arcsys includes a module that is capable of dynamically calling several algorithms. The choice of an algorithm type is valid for all archived objects within the same Arcsys product version; compatibility with algorithms from the previous version is guaranteed. The associated term *hash function* is also used.

Indexing mask

As is the case with the storage policy, an indexing mask is a rule that is referenced by a collection. An indexing mask can be referenced by several collections. An indexing mask refers to the use of a set of Keyword = Value pairs. The keyword component is set to make sense in a specific business application (e.g. Accounting Day, Department, Account No., Account Holder, etc.). The value component can be either unrestricted, or restricted to a set of acceptable values (e.g. A, B or C), or in date format, or restricted by an input mask. Some pairs are defined as mandatory whereas others may be optional.

An application which uses an indexing mask through a collection must supply all Keyword=Value pairs as they are defined using this mask. Any indexing-related errors lead to the record being rejected for conformity. This record is then added to the list of records with errors.

The indexing mask is defined by an administrator via the Arcsys interface or APIs. It is comprised of a set of metadata element definitions.

Journal

A journal is an XML file which contains a list of PREMIS events.

Lot

Arcsys can consolidate several different objects that form a functional set in a client application in the same physical record. It is comprised of different types of objects: files, databases, or any other type of object managed by Arcsys. It is possible to retrieve the entire lot or one of the objects contained in the lot. The MOREQ2010 record is translated in Arcsys implementation by a lot; the lot, as opposed to a MOREQ2010 record, can represent documents that are not yet archived.

Lot enrichment

The process of adding new objects to an existing archive.

Manifest

The manifest is an XML file that defines precisely the content of a record. The manifest contains: metadata associated with the record, structure metadata, a description of the physical files of the record(s) that follow, the object-by-object content of the record, object formats, object names, their size, hash value, the algorithm used to calculate the hash value, etc. The manifest is a type of complete ID card for the record.

The manifest is always written on the storage media and precedes the record that it describes. This process is used to automatically describe storage media (irrespective of the medium). With this system, users can understand media content and metadata without installing the software that generated the records.

Metadata element definition (or keyword)
Component of an indexing mask. We use the term "metadata element definition" rather than the term "keyword" as it is closer to MOREQ2010. The metadata element definition in particular defines the type of metadata (date, string, digital, controlled) and its input mask, for example.
See Also **Indexing mask**.

Object
The object is a basic archived unit that can be retrieved via Arcsys. Lots contain one or more objects. An object can be: a file, a directory, a table, a relational table, etc. The MOREQ2010 component is implemented by this object concept; the object, as opposed to a MOREQ2010 component, can represent a document that has not yet been archived.

Online
Storage level, which must be disk type, that makes records permanently available within an extremely reduced time period.

Permissions
Permissions refer to the user profiles or groups authorized to access documents or sets of documents archived in the system.

Program exit
Place in the standard workflow for picking up and executing specific code.
See Also **Workflow**.

Proof folder
A proof folder consists of a record, a proof slip, and, where appropriate, additional items (common signature or timestamp response, for example) that are used, by demonstrating the fixity and the authenticity of a document, for admission as proof. A proof slip can be generated using Arcsys Web Agent, ArcWeb Module, or Arcsys REST API. A proof folder can only be generated using ArcEP.

Record
A record is an evidential document that is deemed sufficiently important by the creator to be managed by an ERM that will manage its life cycle (retention, disposal, etc.). A record represents an archived lot. A record is archived via a *record request*. Archiving a document *creates a record*.

Relational database (or referential)
Essential component of the system, it contains all the data (excluding archived data) used by Arcsys for its operation. It includes logical entities called "repositories" (see definition).

Repository
Logical entity in the Arcsys relational database. The company can define as many repositories as it wants, either to define a test set, to isolate an application, or for any other reason. These repositories are entirely independent of each other. They all have their own pattern and all have the same structure.

Restore(or retrieval)
This term is used misleadingly in Arcsys to refer to the concept of archive retrieval. It is not accepted in archiving terminology as to mean transfer and then destruction.
See Also **Archive Retrieval**.

Retention and disposal schedule
This comprises all the rules defining the record retention period for a company or an organization, according to risks of unavailability and information system access requirements. It specifies the disposal after these time periods.
See Also **Retention schedule**.

Retention period
A duration expressed in days, months or years of object retention. The retention period is a concept used notably in MOREQ2010.

Retention schedule
A retention schedule defines the start and the end of the retention of records that are attached to it, either directly or through their class.

Retention start date
Date from which a retention period must be taken into account. The retention start date is a concept used notably in MOREQ2010.

Security
An ERMS requirement that involves including documents whose use (confidentiality, risk of exposure) and/or fixity (non modification of content, non-alteration of media) should be closely monitored.

Storage policy
A storage policy is a rule that is referenced by a collection. The policy dictates the storage media which are successively implemented to hold a record, as well as the retention period for each media. The storage policy is defined through the graphical interface. Applications or business users use it indirectly through the reference to a collection. A storage policy can be changed over time to reflect new retention periods or new storage media. The policy covers storage units by logical pool.

Storage pool
Logical storage pool, characterized in particular by its time period (e.g. 10 years). The storage policy assigns a "zone" to a "policy".

Storage zone
The storage zone is a logical entity representing a physical storage space (e.g. set of file systems, tape libraries, cloud storage).

Synchronous retrieval
Archive retrieval that takes place in the form of a direct retrieval of a document (for direct viewing or downloading) in a Web browser.

See Also **Archive Retrieval**.

Time stamping

Time stamping is a technique used to associate a document with a certain date in reference to a given and recognized time system. The date set in this way is an essential element for document authentication. Time stamping can be performed internally or by a third-party time stamp.

Tracking

Result of continuously creating, capturing and maintaining information about the movement and use of the system and objects (ISO 15489-1:2001, 3.19).

Transfer

In an archival sense, this operation sends an archived object to another IT system. Once the transfer is performed, the object can be removed from the ERMS as needed. In OAIS terminology, a transfer represents more specifically the physical transmission of a record or a set of records by a service supplying an archive service. Not to be confused with the transfer of data in the purely technical sense, as with the Arcsys Transfer Server module.

Transit Zone

Entity logically connecting an application agent and a directory, along with additional configuration.

Workflow

A set of operations carried out from the beginning to the end of a process. In Arcsys, this refers to all actions carried out on archives and objects, either directly or indirectly, in the case of archives, from their pre-archiving or preparation to their removal from the system (after they have reached end-of-life). There are other workflows in Arcsys besides the archiving workflow, which are more administration-oriented. Customized workflow involves the use of at least one drop-off point to carry out customer processing.

Registered Trademarks

Firefox is a registered trademark of the Mozilla Foundation.

Linux is a registered trademark of Linus Torvalds.

UNIX is a registered trademark of the Open Group.

Microsoft Windows, Windows NT, Windows Server, SQL Server, Internet Explorer are registered trademarks of Microsoft Corporation in the United States and/or other countries.

SAP is a registered trademark of SAP AG in Germany and other countries.

MySQL is a registered trademark of Oracle and/or its subsidiaries. MariaDB is a registered trademark of Monty Program Ab.

Java is a registered trademark of Oracle and/or its subsidiaries in the United States and other countries.

Infotel is a registered trademark of Infotel SA.

All trademarks mentioned are the property of their respective owners.

Infotel Technical Support

https://techsupport.infotel.com