

Adabas Auditing on z/OS

Installation

Version 2.3.1

November 2024

This document applies to Adabas Auditing for z/OS Version 2.3.1 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2020-2024 Software GmbH, Darmstadt, Germany and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software GmbH product names are either trademarks or registered trademarks of Software GmbH and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software GmbH and/or its subsidiaries is located at <https://softwareag.com/licenses>.

Use of this software is subject to adherence to Software GmbH's licensing conditions and terms. These terms are part of the product documentation, located at <https://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software GmbH Products / Copyright and Trademark Notices of Software GmbH Products". These documents are part of the product documentation, located at <https://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software GmbH.

Document ID: LA-INSTALL-231-20241115EN

Table of Contents

Installation	v
1 About this Documentation	1
Document Conventions	2
Online Information and Support	2
Data Protection	3
2 System Requirements	5
Supported Operating System Platforms	6
Supported Hardware	7
Adabas Requirements	7
Adabas Online System (AOS) Requirements	7
Entire Net-Work Requirements	8
Natural Requirements	8
Predict Requirements	8
LUW Application Requirements	9
Licensing Requirements	9
3 About the Installation Tape	11
Data Sets Delivered	12
Copying the Tape Contents	12
4 Installation for z/OS	15
Step 1. Install the Adabas Audit Load Modules	16
Step 2. Apply Necessary Zaps	16
Step 3. Check, Prepare, and Install the Product License File	16
Step 4. Allocate and Format the Adabas Audit Server	18
Step 5. Load the Auditing System File	18
Step 6. (Optional) Load an SLOG System File	19
Step 7. Load the Adabas Auditing Configuration	20
Step 8. Set Up the Adabas Audit Server Start-up JCL	20
Step 9. Start the Adabas Audit Server	21
5 Installation for z/OS using z/OSMF Workflows	23
Preliminary Installation Steps	24
Workflow Installation Steps	25
6 Activating Auditing for Application Files	37
Step 1. Supply Auditing Definitions to the Audit Server	38
Step 2. Enable Database Auditing with ADARUN AUDITING=YES	38
Step 3. Identify Files for Auditing by an Audit Name	38
Step 4. Enable File Auditing using Database ADAANP statements	39
7 Activating Auditing for TP Monitors	41
Installing Adabas Auditing under Com-plete	42
Installing Adabas Auditing under CICS	43
Installing Adabas Auditing under Batch/TSO	43
Installing Adabas Auditing under IMS/TM	44
Adabas Auditing and Adabas Review	45
8 Activating Auditing for LUW applications	47

Enable LUW applications for auditing	48
--	----

Installation

This chapter describes the prerequisites and installation procedure for installing Adabas Auditing.

System Requirements	Describes the system requirements of Adabas Auditing.
About the Installation Tape	Describes the installation tape.
Installation for z/OS	Provides steps in installing Adabas Auditing on z/OS systems.
Installation for z/OS using z/OSMF Workflows	Provides steps in installing Adabas Auditing on z/OS systems using z/OSMF Workflows.
Activate Auditing for Application Files	Gives instructions how to activate Auditing for Application Files.
Activate Auditing for TP Monitors	Describes how to activate Adabas Auditing for TP monitors.
Activating Auditing for LUW applications	Describes how to enable LUW applications for auditing.

1

About this Documentation

■ Document Conventions	2
■ Online Information and Support	2
■ Data Protection	3

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Monospace font	Identifies service names and locations in the format <i>folder.subfolder.service</i> , APIs, Java classes, methods, properties.
<i>Italic</i>	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
Monospace font	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the symbol.
[]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Product Documentation

You can find the product documentation on our documentation website at <https://documentation.softwareag.com>.

Product Training

You can find helpful product training material on our Learning Portal at <https://learn.software-ag.com>.

Tech Community

You can collaborate with Software GmbH experts on our Tech Community website at <https://tech-community.softwareag.com>. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software GmbH news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at <https://github.com/softwareag> and <https://containers.softwareag.com/products> and discover additional Software GmbH resources.

Product Support

Support for Software GmbH products is provided to licensed customers via our Empower Portal at <https://empower.softwareag.com>. Many services on this portal require that you have an account. If you do not yet have one, you can request it at <https://empower.softwareag.com/register>. Once you have an account, you can, for example:

- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

Data Protection

Software GmbH products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

2 System Requirements

■ Supported Operating System Platforms	6
■ Supported Hardware	7
■ Adabas Requirements	7
■ Adabas Online System (AOS) Requirements	7
■ Entire Net-Work Requirements	8
■ Natural Requirements	8
■ Predict Requirements	8
■ LUW Application Requirements	9
■ Licensing Requirements	9

This section describes the system requirements of Adabas Auditing.

Supported Operating System Platforms

Software AG generally provides support for the operating system platform versions supported by their respective manufacturers. When an operating system platform provider stops supporting a version of an operating system, Software AG will stop supporting that version.

For information regarding Software AG product compatibility with IBM platforms and any IBM requirements for Software AG products, please review the [Product Compatibility for IBM Platforms web page](#).

Before attempting to install this product, ensure that your host operating system is at the minimum required level. For information on the operating system platform versions supported by Software AG products, complete the following steps:

1. Access Software AG's Empower web site at <https://empower.softwareag.com>.
2. Log into Empower.
3. Once logged in, you can expand Products & Documentation in the left menu of the web page and select Product Version Availability to access the Product Version Availability screen.
4. Use the fields on the top of this screen to filter its results for your Software AG product. Click the Search button and the supported Software AG products will be listed in the table below the filter criteria. This list provides, by supported operating system platform:
 - the Software AG general availability (GA) date of the Software AG product
 - the date the operating system platform is scheduled for retirement (OS Retirement)
 - the Software AG end-of-maintenance (EOM) date for the product
 - the Software AG end-of-sustained-support (EOSS) date for the product.



Note: Although it may be technically possible to run a new version of your Software AG product on an older operating system, Software AG cannot continue to support operating system versions that are no longer supported by the system's provider. If you have questions about support, or if you plan to install this product on a release, version, or type of operating system other than those listed on the Product Version Availability screen described above, consult Software AG technical support to determine whether support is possible and under what circumstances.

Supported Hardware

For general information regarding Software AG product compatibility with other platforms and their requirements for Software AG products, visit Software AG's [Hardware Supported](#) web page.

Adabas Requirements

This section lists the Adabas related requirements for successful operation of Adabas Auditing.

- [Adabas Version Requirements](#)
- [ADARUN Requirements](#)

Adabas Version Requirements

The Adabas Audit Server must run with a minimum Adabas version of 8.5 SP1 or, if the Adabas version is greater than 8.5 SP1, the same (or later) version of Adabas as the Adabas database(s) whose data is being audited.

Adabas databases whose data is to be audited must run with Adabas version 8.4 SP2 or later.

In all cases, the Adabas version must be updated with the appropriate version specific zaps from the `ALAvrs.MVSZAPS` data set and any subsequent `ALAvrs.MVSZnnn` data sets, if they have been provided.

ADARUN Requirements

Adabas Auditing requires some ADARUN parameter settings. For more information, refer to the [Installation for z/OS](#) section below.

Adabas Online System (AOS) Requirements

A licensed copy of Adabas Online System (AOS) or the demo version of AOS are not required to support Adabas Auditing.

However, AOS offers some useful statistics and help with the planning and administrative functions of Adabas Auditing. If you choose to install it, be sure to follow the installation instructions in the AOS manual for licensed versions.

If you only use a demo copy of AOS, have in mind that:

- Only limited information concerning Adabas Auditing is available to you with the demo copy.

- The same versions and maintenance levels of the AOS demo code are required as for a licensed copy of AOS.

Entire Net-Work Requirements

1. When you use Entire Net-Work Administration (or Entire Net-Work for Mainframes) and Entire Net-Work Client (or Entire Net-Work for Open Systems), the Adabas Audit Servers and Adabas databases you maintain must be UES-enabled.
2. As an alternative to running a separate Net-work session, the Adabas Audit Server can run with ADATCP. For more information, see the *Entire Net-Work Administration* documentation > *Adding Targets*. For further details on ADATCP, refer to the *Entire Net-Work Administration* documentation > *Enabling Direct TCP/IP Access (ADATCP) to Your Adabas Nucleus*.

If the appropriate Entire Net-Work mainframe and client products are not already installed on your systems, install Entire Net-Work Administration on the mainframe and Entire Net-Work Client on the client side. For complete information on these products, please refer to the section *Entire Net-Work Client Administration* and the *Administration* section of the respective platform's *Entire Net-Work* documentation.

Natural Requirements

Natural 8.2 SP7 or later is recommended when using the Adabas Auditing Configuration to set up Auditing definitions. Auditing is independent of your version of Natural and is compatible with supported Natural levels.

Predict Requirements

Should you decide to use the Adabas Auditing feature that allows you to generate format buffers and a field table using Predict, you must also install a supported version of Predict.

LUW Application Requirements

For LUW applications to be audited, the following are the minimum release levels required:

- WCL 1.8.1.1
- ACL 7.1

Licensing Requirements

In order to execute correctly, Adabas Auditing requires `MLC136.LOAD` or higher load library to be installed regardless of the version of the Adabas nucleus. Refer to [Installation for z/OS](#) for information regarding the Adabas Auditing product license and its requirements.

3

About the Installation Tape

■ Data Sets Delivered	12
■ Copying the Tape Contents	12

The section describes the installation tape.

Data Sets Delivered

This section describes the data sets that are delivered with Adabas Auditing. In all data set names *vrs* represents the version, release, and maintenance level numbers of the release of Adabas Auditing. The following data sets are delivered with Adabas Auditing for z/OS:

Data Set Name	Contains
ALAvrs.INPL	INPL of the Adabas Auditing Configuration
ALAvrs.SYSF	Base Auditing System File
ALAvrs.JOBS	Sample JCL
ALAvrs.LOAD	Load modules
ALAvrs.ZAPS	Zaps for the support of Adabas Auditing

Copying the Tape Contents

Before you perform the individual installation procedure for each component, copy the data sets from the supplied installation medium to your disk.

The way you copy the data sets depends on the installation method and the medium used:

- If you use System Maintenance Aid (SMA), refer to the copy job instructions provided in the *System Maintenance Aid* documentation.
- If you are not using SMA and want to copy the data sets from CD-ROM, refer to the *README.TXT* file on the CD-ROM.
- If you are not using SMA and want to copy the data sets from tape, follow the instructions in this section.

The following steps explain how to copy all data sets from tape to disk:

- [Step 1: Copy Data Set COPY.JOB from Tape to Disk](#)
- [Step 2: Modify hilev.COPY.JOB on Your Disk](#)

■ Step 3: Submit COPY.JOB

Step 1: Copy Data Set COPY.JOB from Tape to Disk

1. Modify the following sample job according to your requirements:

```
//SAGTAPE JOB SAG,CLASS=1,MSGCLASS=X
//* -----
//COPY EXEC PGM=IEBGENER
//SYSUT1 DD DSN=COPY.JOB,
// DISP=(OLD,PASS),
// UNIT=(CASS,,DEFER),
// VOL=(,RETAIN,SER=tape-volser),
// LABEL=(2,SL)
//SYSUT2 DD DSN=hilev.COPY.JOB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=3390,VOL=SER=disk-volser,
// SPACE=(TRK,(1,1),RLSE),
// DCB=*.SYSUT1
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//
```

where:

tape-volser is the VOLSER of the tape, for example: T12345

hilev is a valid high-level qualifier

disk-volser is the VOLSER of the disk.

2. Execute the job to copy the data set COPY.JOB to your disk.

Step 2: Modify hilev.COPY.JOB on Your Disk

1. Modify hilev.COPY.JOB according to your requirements.
2. Set EXPDT to a valid expiration date, for example, 99365.
3. Set HILEV to a valid high-level qualifier, for example, USERLIB.
4. Set LOCATION to a storage location, for example, STORCLAS=ABC or UNIT=3390,VOL=SER=USR123.

Step 3: Submit COPY.JOB

Execute `hi lev.COPY.JOB` to copy single, multiple, or all data sets to your disk.

4

Installation for z/OS

■ Step 1. Install the Adabas Audit Load Modules	16
■ Step 2. Apply Necessary Zaps	16
■ Step 3. Check, Prepare, and Install the Product License File	16
■ Step 4. Allocate and Format the Adabas Audit Server	18
■ Step 5. Load the Auditing System File	18
■ Step 6. (Optional) Load an SLOG System File	19
■ Step 7. Load the Adabas Auditing Configuration	20
■ Step 8. Set Up the Adabas Audit Server Start-up JCL	20
■ Step 9. Start the Adabas Audit Server	21

Be sure your system meets the requirements described in the section [System Requirements](#) above.

To install Adabas Auditing, complete the following steps:



Note: The steps in this section will allow you to install Adabas Auditing, but they will not implement or start Auditing. How Auditing is implemented varies by site, but additional implementation and activation steps are described in the sections [Activate Auditing for Application Files](#) and [Activating Auditing for TP Monitors](#).

Step 1. Install the Adabas Audit Load Modules

Install the Auditing load modules by concatenating the delivered load library in sequence or copying the modules to the library used for running the Adabas nucleus and utilities. The Adabas Audit load modules can be found in `ALAvrs.LOAD`.

Step 2. Apply Necessary Zaps

If they are provided, apply any necessary zaps, as described in the `$README` file on the `ALAvrs.MVSZAPS` library and in any subsequent `ALAvrs.MVSZnnn` data sets.

Step 3. Check, Prepare, and Install the Product License File

You must install a valid license file on all mainframe platforms in which your Software AG mainframe product is installed. The license file is provided as an XML document encoded in US-ASCII. It must remain in that format even on the mainframe and must not be modified. Any modification of the license file will invalidate the digital signature and the license check will fail. In the event of a check failure, please contact your Software AG technical support representative.

For a full product list of license file names, load modules and DD/Link names, refer to the *Adabas Installation* documentation.



Note: Forty days before the license expires (thirty if your MLC is version 1.3.8 or lower), you will receive license check failure messages. Your software product will still function, but these messages warn you that it is time to obtain a new license.

In this step you will prepare the license file, and then install it:

- [Preparing the Product License File](#)

■ Installing the Product License File

Preparing the Product License File

The product license file is supplied on the individual customer installation tape or separately via an e-mail attachment. Before you can install the license, you must transfer it from e-mail or the installation tape and store it on a z/OS system. This section describes how to do this for a license distributed either by e-mail or on the installation tape.

➤ To prepare the license file from an e-mail attachment, complete the following steps:

- 1 Transfer the license to z/OS, as described in the *Adabas for Mainframes* documentation > *Software AG Mainframe Product Licensing* > *Transferring a License File from PC to a z/OS Host Using FTP*.
- 2 Verify that the transferred license file is stored in an Adabas source library (with RECFM=F or RECFM=FB and LRECL=80), making sure to preserve its format as ASCII.

➤ To prepare the license file from the installation tape, complete the following step:

- Verify that the license file is stored in an Adabas source library (with RECFM=F or RECFM=FB and LRECL=80), making sure to preserve its format as ASCII.

Installing the Product License File

Once the license file has been prepared, you can install it in one of two ways:

- Convert the license to a load module (ALALIC) that is then loaded by the Adabas Audit Server.
- Reference the license file in the Adabas Audit Server start-up job by DD statement.

This section describes both methods.

➤ To convert the license file to a load module, complete the following steps:

- 1 Review and modify sample job ASMLICAM in the Adabas JOBS library, as follows:
 - Set the variable MLCLOAD to point to the license load library (MLCVRS.LOAD).

Note: For information regarding the required version of the license load library, refer to the Installation and License section of the release notes relevant to the installed version of Adabas.
 - Set the variable USRLOAD to point to an appropriate user load library.

Note: This user load library must also be included in the STEPLIB concatenation for the Adabas Audit Server.

- Set the LICFILE parameter to point to the data set containing the Adabas Audit Server license file you transferred to z/OS earlier.
 - Set the LMOD parameter to the load module name ALALIC.
- 2 Submit the sample job ASMLICAM. This job runs the MAKE function of the LICUTIL utility to convert the license text file to an assembler source module. ASMLICAM then links and assembles the assembler source to generate a load module called ALALIC which is stored in the specified user load library (USRLOAD). For more information about the LICUTIL utility, read the *Adabas for Mainframes* documentation > *Software AG Mainframe Product Licensing* > *Using The License Utility: LICUTIL*.
 - 3 Update your Adabas Audit Server start-up job to reference the user load library so ALALIC will be loaded by the Adabas Audit Server, as described in [Step 8: Set Up the Adabas Audit Server Start-up JCL](#).

➤ To reference the license file in the Adabas Audit Server start-up job, complete the following steps:

- 1 Make sure any previously created ALALIC load module is inaccessible to the Adabas load library being used by your Adabas Audit Server job. The Audit Server first tries to load ALALIC and, if unsuccessful, it reads from a dataset defined to the DD statement DDLALA.
- 2 Update your Adabas Audit Server start-up jobs to reference the license, as described in [Step 8: Set Up the Adabas Audit Server JCL](#).

Step 4. Allocate and Format the Adabas Audit Server

Create a database to use as your Audit Server. Refer to *Reference > Utilities Used with Adabas Auditing > ADADEF MODIFY AUDITSERVER Function* or *ADADEF DEFINE AUDITSERVER Function* for an explanation of the parameters used for defining an Audit Server.



Note: User application files should not be loaded on the Adabas Audit Server.

Step 5. Load the Auditing System File

To use Adabas Auditing and customize its processing, you must supply various Auditing definitions.

You maintain your Auditing definitions in the Auditing system file using the Adabas Auditing Configuration. The Adabas Auditing Configuration is an online interface that must be loaded into Natural before you can use it. Access to the Adabas Auditing Configuration is then available through Natural. For more information, refer to the *Adabas Auditing Configuration* chapter.

To maintain your Auditing definitions in the Auditing system file, you must first load the Auditing system file on the Adabas Audit Server.

➤ **To load an Auditing system file on the Adabas Audit Server:**

- 1 Use the ADALOD utility. A sample ADALOD utility job is provided in member ADALODSF of the `ALAvrs.JOBS` data set. Refer to *Reference > Utilities Used with Adabas Auditing > ADALOD LOAD Parameters for the Adabas Audit Server* for an explanation of the parameters used for loading the Auditing system file. You need to specify the AUDITING parameter on the ADALOD LOAD function:

```
ADALOD LOAD FILE=nnnn,AUDITING
```

where `nnnn` is the number of the Auditing system file.

You have to also specify the supplied Auditing system file dataset `ALAvrs.SYSF` on the DDE-BAND input DD name.

- 2 Do either of the following in Natural to identify the physical database file to be associated with the system file:
 - Run a Natural NTLFILE macro for the file. For complete information, refer to Software AG's [Empower web site](#) > *Natural for Mainframes > Natural LFILE parameter* documentation.
 - When you start the Adabas Auditing Configuration ensure an LFILE parameter is specified either as a dynamic parameter or inside a Natural SYSPARM profile.

Regardless of which method you use, the LFILE setting should be:

```
LFILE=(89,dbid,file)
```

where `dbid` is the database ID of the Adabas Audit Server you created during installation and `file` is the Auditing system file you have loaded using ADALOD.

Step 6. (Optional) Load an SLOG System File

If you intend to use the subscription logging (SLOG) facility, you will need an Auditing SLOG system file which is an Adabas system file on the Adabas Audit Server. Refer to the *Operations chapter > Using the Subscription Logging Facility > Setting up Subscription Logging* for an explanation of how to set up the Auditing SLOG facility.

Step 7. Load the Adabas Auditing Configuration

Adabas Auditing Configuration is used to create and maintain Auditing definitions in the Auditing system file. You have to load it into Natural. To do this, use Natural's INPL utility to load the `ALAvrs.INPL` data set (and any supplied `ALAvrs.IXnn` updates) into the Natural system file libraries.

Step 8. Set Up the Adabas Audit Server Start-up JCL

Set up the start-up JCL and parameters for the Adabas Audit Server job. Use the JCL for an existing Adabas nucleus as a starting point for creating the Audit Server JCL. A sample Audit Server start-up job is provided in member `ADANUC2` of the `ALAvrs.JOBS` data set.

Note the specification of the required Audit Log datasets `DDALOGRn` and the Audit Error dataset `DDAUDERR` as shown in the `ADANUC2` member.

If the `DDAUDERR` file is directed to a dataset it will be given the DCB attributes `RECFM=FB`, `LRECL=80`. Messages on `DDAUDERR` are formatted similarly to a multi-line WTO: the message number and DBID appear on only the first line of the message. Each line is prefixed with the Julian date and local time.

Refer to the *Reference* chapter > *Pertinent ADARUN Parameters for Auditing* for an explanation of the ADARUN parameters necessary for successful operation of Adabas Audit Server. Recommended values for some ADARUN parameters are as follows:

1. Set ADARUN parameter `LAP` to the size of the Auditing buffer. For performance reasons Software AG recommends setting `LAP` to a relatively large value (e.g. `LAP=40M`).
2. Set ADARUN parameter `LU` to 167,000 or greater.
3. Set ADARUN parameter `NAB` to a value greater than or equal to:

```
41 * 10 * the-number-of-Adabas-nuclei-sending-data-to-the-Adabas-Audit-Server
```

For example, if one Adabas nucleus will be sending data to the Adabas Audit Server, set the `NAB` parameter greater than or equal to 410 (for example `NAB=420`).

4. Set ADARUN parameter `NT` to a value greater than or equal to 15.

You must also make the following license-related modifications:

- Software AG licensing requires that the modules `LICMAIN` and `LICUTIL` be loaded when your Audit Server starts up. These modules are distributed in the `MLCvrs.LOAD` library. You must either:
 - Copy `LICMAIN` and `LICUTIL` into `ALAvrs.LOAD` or

- Concatenate `MLCvrs.LOAD` with `ALAvrs.LOAD`.
- Verify that the appropriate product license file is correctly referenced in the Audit Server job. Do either of the following:
 - Verify that the license load module installed in [Step 3: Check, Prepare, and Install the Product License File](#) is stored in a load library that is accessible to the Audit Server job. Add the user load library in which the license load module resides to the STEPLIB concatenation of the Audit Server job or copy the licensing module into `ALAvrs.LOAD` or
 - Verify that there is no ALALIC module accessible to the Audit server job and that the following DD statement is included in the Audit Server job:

```
//DDLALA DD DISP=SHR,DSN=dsn
```

where *dsn* is the data set name of the license file loaded from the tape in ASCII format. Note that *dsn* could reference a member in a partitioned data set.



Note: Adabas first tries to load the license load module ALALIC and, and if unsuccessful, it reads from the DDLALA data set.

Step 9. Start the Adabas Audit Server

Run the Adabas Audit Server job you set up in the previous step. A sample Audit Server job is provided in member `ADANUC2` of the `ALAvrs.JOBS` data set.

5

Installation for z/OS using z/OSMF Workflows

■ Preliminary Installation Steps	24
■ Workflow Installation Steps	25

This section describes an alternative means of installing Adabas Auditing using z/OSMF workflows.

Before starting, be sure your system meets the requirements described in the section [System Requirements](#).

To install Adabas Auditing using z/OSMF workflows, complete the following [Preliminary Installation Steps](#) and [Workflow Installation Steps](#).



Note: The steps in these sections will allow you to install Adabas Auditing but how Auditing is implemented varies by site. Additional implementation and activation steps are described in the sections [Activate Auditing for Application Files](#) and [Activating Auditing for TP Monitors](#).

Preliminary Installation Steps

- [Step 1. Install the Adabas Auditing Load Modules](#)
- [Step 2. Apply Necessary Zaps](#)
- [Step 3. Check and Prepare the Adabas Auditing License File](#)
- [Step 4. Confirm Availability of the Adabas License File](#)
- [Step 5. Create the Adabas Auditing Workflow Instance](#)

Step 1. Install the Adabas Auditing Load Modules

Refer to [About the Installation Tape](#) for information on the Adabas Auditing data sets delivered and a sample job to copy them to disk. The Adabas Auditing load modules are delivered in `ALAvrs.LOAD`.

Step 2. Apply Necessary Zaps

If provided, apply any necessary zaps as described in the `$README` file on the `ALAvrs.ZAPS` library and in any subsequent `ALAvrs.Znnn` data sets.

Step 3. Check and Prepare the Adabas Auditing License File

Verify the availability of your Adabas Auditing license file and refer, as necessary, to the Installation and License section of your current Adabas for Mainframes release notes for information on how to prepare the license file in a sequential data set of `RECFM=FB` and `LRECL=80`. The workflow installation will require that the name of this sequential data set is defined to the `DSN-ALA-LICS` variable in [Workflow Step 2. Auditing](#).

Step 4. Confirm Availability of the Adabas License File

The Adabas Audit Server requires an Adabas license. The workflow installation assumes that an Adabas license load module ADALIC resides in the data set identified by the `DSN-ADA-USER-LOAD` variable in [Workflow Step 3. Adabas](#).

If this is not the case, the start-up procedure for the Adabas Audit Server (see [Workflow Step 11. Setup nucleus](#)) must be modified to make an ADALIC license module available via the STEPLIB concatenation or to specify the data set name of an XML license file using the DDLIC DD statement.

Step 5. Create the Adabas Auditing Workflow Instance

The Adabas Auditing z/OSMF workflow files can be downloaded from the Software Download Center in [Empower](#). Log in to Empower, expand **Products & Documentation** on the left and select **Download Products** to access the Software Download Center.

The Adabas Auditing z/OSMF workflow files consist of:

File Name	Contains
<code>ALAvrs-install.xml</code>	Workflow definition file
<code>ALAvrs.install.properties</code>	Workflow variable input file

Transfer the workflow definition file `ALAvrs-install.xml` and the workflow variable input file `ALAvrs-install.properties` to the UNIX System Services filesystem or a partitioned data set (RECFM=FB and LRECL=1024), ensuring that the files are stored in EBCDIC (IBM-1047) format.

Refer to the IBM z/OSMF documentation for information on how to use the z/OSMF Workflows task (not the Workflows Editor), available from the z/OSMF desktop, to create a workflow by importing the Adabas Auditing workflow files listed above.

By default, the name of the imported Workflow will be `Adabas Auditing v.r.s.`

Workflow Installation Steps

After successfully completing the [Preliminary Installation Steps](#), use the z/OSMF Workflows page in the z/OSMF Workflows task to view and manage the imported Adabas Auditing workflow.

From the z/OSMF Workflows page, select and open the Adabas Auditing workflow to display the workflow Steps page. This page lists all the workflow steps necessary to install Adabas Auditing.

Each step is executed by clicking on the step name and selecting the **Perform** tab. Depending on the step's function, execution of the step may require values to be assigned to specific variables (for use by the current step and/or subsequent steps), and confirmation of correctly generated JCL.

Progress through the workflow steps by executing each step in order, detailed information regarding each of the workflow steps is provided below.

- [Workflow Step 1. Product data sets](#)
- [Workflow Step 2. Auditing](#)
- [Workflow Step 3. Adabas](#)
- [Workflow Step 4. Natural](#)
- [Workflow Step 5. System](#)
- [Workflow Step 6. Create user libraries](#)
- [Workflow Step 7. Prepare license key](#)
- [Workflow Step 8. Define and format server](#)
- [Workflow Step 9. Load System File](#)
- [Workflow Step 10. Load SLOG file](#)
- [Workflow Step 11. Setup nucleus](#)
- [Workflow Step 12. APF authorization](#)
- [Workflow Step 13. Start server](#)
- [Workflow Step 14. Check server](#)
- [Workflow Step 15. Assemble link routines](#)
- [Workflow Step 16. Natural INPL](#)
- [Workflow Step 17. Activate Auditing](#)

Workflow Step 1. Product data sets

This step configures the default data set names for Adabas Auditing, Adabas, Mainframe License Check utility, and Natural.

Select the **Perform** tab to define values for the following variables:

Variable Name	Description	Default Value
LIB-GROUP	The High-Level Qualifier for the Adabas Auditing data sets created in Preliminary Installation Step 1 .	SAGLIB
VER-ADA	The version of Adabas installed on your system in <i>ADA_{vrs}</i> format.	ADA _{vrs}
VER-ALA	The version of Adabas Auditing being installed in <i>ALA_{vrs}</i> format.	ALA _{vrs}
VER-MLC	The version of the Mainframe License Check utility installed on your system in <i>MLC_{vrs}</i> format.	MLC _{vrs}
VER-NAT	The version of Natural installed on your system in <i>NAT_{vrs}</i> format.	NAT _{vrs}

On completion, this workflow step will generate the default data set names for the ADA, ALA, MLC and NAT product libraries installed on your system by concatenating LIB-GROUP with VER-ADA, VER-ALA, VER-MLC and VER-NAT respectively.



Note: If your site does not follow this naming convention, the generated product data set names can be subsequently modified in Workflow Steps 2, 3 and 4.

Workflow Step 2. Auditing

This step enables you to confirm, and optionally modify, the names of the Adabas Auditing data sets generated by [Workflow Step 1. Product data sets](#).

Select the **Perform** tab to display the relevant variable names and their corresponding generated default data set names as described by the following table:

Variable Name	Description	Generated Default Data Set Name
DSN-ALA-SYSF	The Adabas Auditing system file.	<LIB-GROUP>.<VER-ALA>.SYSF
DSN-ALA-LOAD	The Adabas Auditing LOAD library.	<LIB-GROUP>.<VER-ALA>.LOAD
DSN-ALA-INPL	The Adabas Auditing INPL.	<LIB-GROUP>.<VER-ALA>.INPL
DSN-ALA-LICS	The Adabas Auditing license file sequential data set.	<LIB-GROUP>.<VER-ALA>.LICS
DSN-ALA-ISNN	* Adabas Auditing INPL update, if any.	None

* The DSN-ALA-ISNN variable is optional. If the installation media contained INPL updates, specify the name of the highest numbered INPL update data set, otherwise leave the field blank.

Before completing this workflow step, ensure the specified data set names are correct for your system, if not, modify them accordingly.

Workflow Step 3. Adabas

This step enables you to confirm, and optionally modify, the names of the Adabas data sets generated by [Workflow Step 1. Product data sets](#).

Select the **Perform** tab to display the relevant variable names and their corresponding generated default data set names as described by the following table:

Variable Name	Description	Generated Default Data Set Name
DSN-ADA-LOAD	The Adabas LOAD library.	<LIB-GROUP>.<VER-ADA>.LOAD
DSN-ADA-SRCE	The Adabas SRCE library.	<LIB-GROUP>.<VER-ADA>.SRCE
DSN-ADA-USER-LOAD	The LOAD library where the Adabas license load module ADALIC resides.	<LIB-GROUP>.<VER-ADA>.LOAD
DSN-MLC-LOAD	The Mainframe License Check load library.	<LIB-GROUP>.<VER-MLC>.LOAD

Before completing this workflow step, ensure the specified data set names are correct for your system, if not, modify them accordingly.

Workflow Step 4. Natural

This step enables you to confirm, and optionally modify, the name of the Natural data set generated by [Workflow Step 1. Product data sets](#) and, in addition, define the Natural environment for use by [Workflow Step 16. Natural INPL](#).

Select the **Perform** tab to display the relevant variable names and their corresponding values as described by the following table:

Variable Name	Description	Default Value
DSN-NAT-LOAD	The Natural LOAD library.	<LIB-GROUP>.<VER-NAT>.LOAD
NAT-PGM	The batch Natural program name.	NATBAT
NAT-PRM	* The batch Natural parameters.	None

* The NAT-PRM variable can be used to pass parameters to Natural. The value specified will be passed instream as the CMPRMIN DD input. If no parameters are required, leave the field blank.

Before completing this workflow step, ensure the specified values are correct for your system, if not, modify them accordingly.

Workflow Step 5. System

This step enables you to define the necessary z/OS system libraries for use by subsequent workflow steps.

Select the **Perform** tab to display the relevant variable names and their corresponding values as described by the following table:

Variable Name	Description	Default Value
DSN-USERPROCLIB	Location for the Adabas Audit Server start-up procedure used by Workflow Step 11. Setup nucleus .	None
DSN-SYMACLIB	System macro library.	SYS1.MACLIB
DSN-SCEELKD	Language edition routines.	SYS1.CEE.SCEELKD

Before completing this workflow step, ensure the specified values are correct for your system, if not, modify them accordingly.

Workflow Step 6. Create user libraries

This step will create the Adabas Auditing user load and source libraries using variables defined in [Workflow Step 1. Product data sets](#).

Select the **Perform** tab to display the relevant variable names and their corresponding values as described by the following table:

Variable Name	Description	Default Value
LIB-GROUP-UNIT	The device unit type to be used (UNIT=).	SYSALLDA
LIB-GROUP-VOLSER	(optional) The device name to be used (VOL=SER=).	None
DSN-ALA-USER-LOAD	The LOAD library to be used for storing the Adabas Auditing license load module ALALIC (Workflow Step 7. Prepare license key) and the audit-enabled Adabas link routines (Workflow Step 15. Assemble link routines).	<LIB-GROUP>.ALA.LOAD
DSN-ALA-USER-SRCE	The SRCE library to be used in the preparation of the license key (Workflow Step 7. Prepare license key), the storing of the LINKGBLS definitions (Workflow Step 15. Assemble link routines) and the location for the Adabas Audit Server configuration parameters (Workflow Step 11. Setup nucleus).	<LIB-GROUP>.ALA.SRCE

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 7. Prepare license key

This step will create the Adabas Auditing license load module ALALIC.

Select the **Perform** tab to display the relevant variable names and their corresponding values as described by the following table:

Variable Name	Description	Default Value
DSN-ALA-LICS	The Adabas Auditing license file sequential data set.	As defined in Workflow Step 2. Auditing .
VER-ALA-LIC	The Adabas Auditing license version.	ALA v. r

The Mainframe License Check utility will be used to convert the license file defined to the variable DSN-ALA-LICS in [Workflow Step 2. Auditing](#) to the load module ALALIC stored in the library defined to the variable DSN-ALA-USER-LOAD in [Workflow Step 6. Create user libraries](#).

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 8. Define and format server

This step will run the Adabas utilities ADAFRM and ADADEF to create the Adabas Auditing container files and define them as an Adabas Audit Server.

Select the **Perform** tab to display the relevant variable names and their corresponding values as described by the following table:

Variable Name	Description	Default Value
DB-ID	The Database ID to be defined to the Audit Server.	None
DB-SVC	The number of the Adabas SVC.	None
DB-DSN-PREFIX	The prefix to be used for the Audit Server container files.	SAG.ADABAS
DB-DEVICE	The device unit type to be used (UNIT=).	3390
DB-VOL	The device name to be used (VOL=SER=).	None

The DB-DSN-PREFIX value will be combined with the DB-ID value to provide the prefix used for the container file data sets. For example, the default container file data set prefix will be SAG.ADABAS.DBnnnnn where nnnnn is the value defined to the variable DB-ID.

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 9. Load System File

This step will run the Adabas utility ADALOD to load the Adabas Auditing system file into the Audit Server container files created by [Workflow Step 8. Define and format server](#).

Select the **Perform** tab to display the relevant variable names and their corresponding values as described by the following table:

Variable Name	Description	Default Value
FNR-SYSF	The File Number to be used for the Adabas Auditing system file in the Audit Server.	None
DSN-ALA-SYSF	The Adabas Auditing system file.	As defined in Workflow Step 2. Auditing .

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 10. Load SLOG file

This step will run the Adabas utility ADALOD to load the SLOG system file into the Audit Server container files created by [Workflow Step 8. Define and format server](#).

Select the **Perform** tab to display the relevant variable names and their default values as described by the following table:

Variable Name	Description	Default Value
FNR-SLOG	The File Number to be used for the SLOG file in the Audit Server.	None

For information about the SLOG system file refer to Using the Auditing Subscription Logging (SLOG) Facility.

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 11. Setup nucleus

This step will establish a procedure for starting the Adabas Audit Server.

Select the **Perform** tab to begin the setup process.

The setup of the Audit Server nucleus follows the conventions used by System Maintenance Aid. The configuration members for the Audit Server and the procedure for the started task are based on the same templates used for an Adabas nucleus.

The start-up procedure for the Audit Server will be placed in member `NUCnnnnn` in the data set defined to the variable `DSN-USERPROCLIB` in [Workflow Step 5. System](#).

The Audit Server `ADARUN` parameters will be placed in member `PRnnnnn` in the data set defined to the variable `DSN-ALA-USER-SRCE` in [Workflow Step 6. Create user libraries](#).

Where `nnnnn` is the value defined to the variable `DB-ID` in [Workflow Step 8. Define and format server](#).

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 12. APF authorization

This step will APF authorize the load libraries used by the Adabas Audit Server.

Select the **Perform** tab to begin the APF authorization process.

The table below identifies those libraries in the STEPLIB concatenation of the Audit Server start-up procedure which must be APF authorized:

Variable Name	Description	Value
DSN-ALA-USER-LOAD	The LOAD library to be used for storing the Adabas Auditing license load module ALALIC.	As defined in Workflow Step 6. Create user libraries.
DSN-ALA-LOAD	The Adabas Auditing LOAD library.	As defined in Workflow Step 2. Auditing.
DSN-ADA-USER-LOAD	The LOAD library where the Adabas license load module ADALIC resides.	As defined in Workflow Step 3. Adabas.
DSN-ADA-LOAD	The Adabas LOAD library.	As defined in Workflow Step 3. Adabas.
DSN-MLC-LOAD	The Mainframe License Check load library.	As defined in Workflow Step 3. Adabas.

This workflow step uses the SETPROG APF command to add the required libraries to your dynamic APF list.

If your site uses a static APF list, or you wish these libraries to maintain their AFP authorization across IPLs, we recommend you discuss these requirements with your systems programmer. Under these circumstances, after the libraries have been added, use the 'Override Complete' option to mark this step as complete.

Otherwise, throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 13. Start server

This step will start the Adabas Audit Server using the procedure established in [Workflow Step 11. Setup nucleus.](#)

Select the **Perform** tab to begin the process of starting the Adabas Audit Server.

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 14. Check server

This step will check the Adabas Audit Server is active by using the Adabas utility ADADBS to issue an `OPERCOM DAUDPARM` command to display the audit parameters in use.

Select the **Perform** tab to begin the process of checking the Adabas Audit Server is active.

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 15. Assemble link routines

This step uses instream `LGBLSET` parameters (including `AUDIT=YES`) to assemble the Adabas link globals table and re-link the Adabas link modules with the appropriate Adabas Auditing stub modules.

Select the **Perform** tab to begin the process of building the audit-enabled Adabas link modules.

The following table describes the variables used in this step:

Variable Name	Description	Value
DSN-ALA-USER-SRCE	The SRCE library used for storing the instream <code>LGBLSET</code> parameters.	As defined in Workflow Step 6. Create user libraries.
DSN-ALA-USER-LOAD	The LOAD library used for storing assembled objects and linked load modules.	As defined in Workflow Step 6. Create user libraries.
DSN-ALA-LOAD	The Adabas Auditing LOAD library.	As defined in Workflow Step 2. Auditing.
DSN-ADA-SRCE	The Adabas SRCE library.	As defined in Workflow Step 3. Adabas.
DSN-ADA-LOAD	The Adabas LOAD library.	As defined in Workflow Step 3. Adabas.

First, the instream `LGBLSET` parameters are copied to members `LNKGBLBA` (for Batch), `LNKGBLTS` (for TSO) and `LNKGBLCO` (for Com-Plete) in the library defined to the variable `DSN-ALA-USER-SRCE`.

Next, these parameter members are assembled to objects of the same name in the library defined to the variable `DSN-ALA-USER-LOAD`. These objects are then linked with the appropriate Adabas Auditing stub module `EABXMVS` (for Batch and TSO) and `EABXCOM` (for Com-Plete).

Finally, these `LNKGBL*` load modules are linked again to create the Adabas link modules `ADALNK` (for batch), `ADALNKTS` (for TSO), `ADALCO` (for Com-Plete), and `ADALNKR` (for re-entrant Batch).



Note: The Adabas Auditing z/OSMF workflow does not currently support the installation of Adabas Auditing under CICS.

Throughout the execution of this step, ensure any variable values are set appropriately for your system, check the instream LGBLSET parameters and review any generated JCL. If necessary, modify accordingly.

For additional information, including installing Adabas Auditing under CICS, refer to [Activating Auditing for TP Monitors](#).

Workflow Step 16. Natural INPL

This step will load the Adabas Auditing INPL.

Select the **Perform** tab to display the relevant variable names and their corresponding values as described by the following table:

Variable Name	Description	Default Value
DSN-ALA-INPL	The Adabas Auditing INPL.	As defined in Workflow Step 2. Auditing .
DSN-ALA-ISNN	Adabas Auditing INPL update, if any.	As defined in Workflow Step 2. Auditing .

In addition, the following variables are also used in this step:

Variable Name	Description	Default Value
DSN-NAT-LOAD	The Natural LOAD library.	As defined in Workflow Step 4. Natural .
NAT-PGM	The batch Natural program name.	As defined in Workflow Step 4. Natural .
NAT-PRM	The batch Natural parameters, if any.	As defined in Workflow Step 4. Natural .
DSN-ADA-LOAD	The Adabas LOAD library.	As defined in Workflow Step 3. Adabas .

Throughout the execution of this step, ensure any variable values are set appropriately for your system and review any generated JCL. If necessary, modify accordingly.

Workflow Step 17. Activate Auditing

This step provides instructions to enable auditing in an Adabas database and activate auditing for an Adabas file.

Select the **Perform** tab to display the relevant variable names and their default values as described by the following table:

Variable Name	Description	Default Value
ADA-DB-ID	The Adabas database number where auditing is to be enabled.	None
ADA-FNR	The Adabas file number to be activated for auditing.	None
ADA-AUDITNM	The Audit name to be assigned to the Adabas file identified by variable ADA-FNR.	None

Modify these values accordingly and continue with the workflow step, following the instructions to enable and activate auditing for an Adabas file.

These instructions are applicable to an Adabas installation made with System Maintenance Aid. If this is not applicable for your site, incorporate the changes listed for the nucleus whose database number is defined by variable ADA-DB-ID.

For additional information, refer to the section [Activating Auditing for Application Files](#).

6

Activating Auditing for Application Files

- Step 1. Supply Auditing Definitions to the Audit Server 38
- Step 2. Enable Database Auditing with ADARUN AUDITING=YES 38
- Step 3. Identify Files for Auditing by an Audit Name 38
- Step 4. Enable File Auditing using Database ADAANP statements 39

Once Adabas Auditing has been installed, the auditing of application files can be implemented as described by the following steps:

Step 1. Supply Auditing Definitions to the Audit Server

Auditing definitions are used to customize the auditing process. This includes specifying the application files that are to be processed for auditing.

These auditing definitions are maintained in the Auditing system file and are read by the Adabas Audit Server at start-up. Refer to *Adabas Auditing Configuration* for information regarding creating and maintaining auditing definitions in the Auditing system file.

Stop and restart the Audit Server to pick up or refresh the auditing definitions in the Auditing system file.

Step 2. Enable Database Auditing with ADARUN AUDITING=YES

To enable database auditing, you must add the ADARUN parameter `AUDITING=YES` to each Adabas nucleus job containing those application files which have been specified in the Auditing definitions used by the Audit Server.

Refer to the *Reference* chapter > *Pertinent ADARUN Parameters for Auditing* for more information on the `AUDITING=` parameter and other auditing related ADARUN parameters appropriate for an Adabas nucleus running `AUDITING=YES`.

Step 3. Identify Files for Auditing by an Audit Name

Each Adabas file participating in auditing must have an audit name. This is a user assigned 8-character value allowing audit information to be tracked. The Audit ID should be unique across all files participating in auditing.

There are 2 ways to assign an audit name to a file:

1. Use the `AUDITNM=` parameter when the file is loaded with ADALOD.
2. Use the MODFCB function of ADADBS with the `AUDITNM=` parameter.

For more information about the use of the `AUDITNM=` parameter with ADALOD LOAD and ADADBS MODFCB, refer to *Reference* > *Utilities Used with Adabas Auditing*.

Step 4. Enable File Auditing using Database ADAANP statements

To implement and activate auditing for application files, you must perform the following:

1. Add the DD name ADAANP to each Adabas nucleus running `AUDITING=YES`.
2. Allocate a sequential file or PDS member to this DD name.
3. For each application file to be audited, add the appropriate for your requirements ADAANP statements to the sequential file or PDS member in the corresponding nucleus JCL.

Refer to the *Reference > Adabas Nucleus Auditing Parameters* for more information about the required ADAANP parameters.

When all necessary modifications have been made, restart the Adabas nuclei.

7

Activating Auditing for TP Monitors

■ Installing Adabas Auditing under Com-plete	42
■ Installing Adabas Auditing under CICS	43
■ Installing Adabas Auditing under Batch/TSO	43
■ Installing Adabas Auditing under IMS/TM	44
■ Adabas Auditing and Adabas Review	45

The following sections describe how to activate auditing for TP monitors.



Note: If you wish Natural-related client information to be included in the auditing data, then you must ensure that the Natural Profile parameter `ADAPRM=ON` is set.

Installing Adabas Auditing under Com-plete

To prepare the Adabas link routine:

1. Copy sample member LCOGBL (provided in the `ADAvrs.SRCE` library) to any appropriate user source library where it can be modified.
2. Modify the LGBLSET parameters in the copied LCOGBL member according to your site requirements ensuring the parameter `AUDIT=` is added and set to YES.
3. Modify sample job ASMGBLS provided in the `ADAvrs.JOBS` library as described at the top of the job. When fully modified, the SET statement in the job should reference the LCOGBL member you prepared in the previous step, and the NAME link edit control statement should reference the name specified by the GBLNAME parameter within that member.
4. Once modified, submit the ASMGBLS job to assemble and link-edit the link globals module. A new link globals module with the name specified by the GBLNAME parameter will be generated in the user load library identified in the ASMGBLS job.
5. Copy sample job LNKLCO8 provided in the `ADAvrs.JOBS` library to a user source library.
6. Modify the sample job LNKLCO8 to link the new link globals module you created along with the following INCLUDE statement:

```
INCLUDE ALALIB(EABXCOM)
```

where ALALIB references the `ALAvrs.LOAD` library.



Note: Instructions for modifying the sample job are described at the top of the job. Be sure to direct the output from the job to an appropriate user load library. The module resulting from this job is called ADALCO.

7. Place the newly linked ADALCO module in a load library available to the Com-plete job.

For general information regarding the installation of Adabas with Com-plete and an explanation of the associated LGBLSET parameters, refer to the *Adabas for Mainframes* documentation > *Installing Adabas with TP Monitors*.

Installing Adabas Auditing under CICS

To prepare the Adabas link routine:

1. Copy sample member CICSGBL (provided in the `ACIvrs.SRCE` library) to any appropriate user source library where it can be modified.
2. Modify the LGBLSET parameters in the copied CICSGBL member according to your site requirements ensuring the parameter `AUDIT=` is added and set to YES.
3. Modify sample job ASMGBLS (provided in the `ADAvrs.JOBS` library) as described at the top of the job. When fully modified, the SET statement in the job should reference the CICSGBL member you prepared in the previous step, and the NAME link edit control statement should reference the name specified by the GBLNAME parameter within that member.
4. Once modified, submit the ASMGBLS job to assemble and link-edit the link globals module. A new link globals module with the name specified by the GBLNAME parameter will be generated in the user load library identified in the ASMGBLS job.
5. Copy sample job LNKGCICS (provided in the `ACIvrs.SRCE` library) to a user source library.
6. Modify the sample job LNKGCICS to link the new link globals module you created along with the following INCLUDE statement:

```
INCLUDE ALALIB(EABXCIC)
```

where ALALIB references the `ALAvrs.LOAD` library.



Note: Instructions for modifying the sample job are described at the top of the job. Be sure to direct the output from the job to an appropriate user load library that will be made available to CICS in the DFHRPL library concatenation.

For general information regarding the installation of Adabas with CICS and an explanation of the associated LGBLSET parameters, refer to the *Adabas for Mainframes* documentation > *Installing Adabas with TP Monitors*.

Installing Adabas Auditing under Batch/TSO

To prepare the Adabas link routine:

1. Copy sample member LNKGBLS (provided in the `ADAvrs.SRCE` library) to any appropriate user source library where it can be modified.
2. Modify the LGBLSET parameters in the copied LNKGBLS member according to your site requirements ensuring the parameter `AUDIT=` is added and set to YES.

3. Modify sample job ASMGBLS (provided in the `ADAvrs.JOBS` library) as described at the top of the job. When fully modified, the SET statement in the job should reference the LNKGBLS member you prepared in the previous step and the NAME link edit control statement should reference the name specified by the GBLNAME parameter within that member.
4. Once modified, submit the ASMGBLS job to assemble and link-edit the link globals module. A new link globals module with the name specified by the GBLNAME parameter will be generated in the user load library identified in the ASMGBLS job.
5. Copy sample job LNKLNK8 (provided in the `ADAvrs.JOBS` library) to a user source library.
6. Modify the sample job LNKLNK8 to link the new link globals module you created along with the following INCLUDE statement:

```
INCLUDE ALALIB(EABXMVS)
```

where ALALIB references the `ALAvrs.LOAD` library.



Note: Instructions for modifying the sample job are described at the top of the job. Be sure to direct the output from the job to an appropriate user load library. The module resulting from this job is called ADALNK.

7. Place the newly linked ADALNK module in a load library available to the Batch/TSO job.

For general information regarding the installation of Adabas with Batch/TSO and an explanation of the associated LGBLSET parameters, refer to the *Adabas for Mainframes* documentation > *Installing Adabas with TP Monitors*.

Installing Adabas Auditing under IMS/TM

To prepare the Adabas link routine:

1. Copy sample member LNIGBL (provided in the `AI Ivrs.SRCE` library) to any appropriate user source library where it can be modified.
2. Modify the LGBLSET parameters in the copied LNIGBL member according to your site requirements ensuring the parameter `AUDIT=` is added and set to YES.
3. Modify and run sample job ASMGBLS (provided in the `ADAvrs.JOBS` library) as described at the top of the job. When fully modified, the SET statement in the job should reference the LNIGBL member you prepared in the previous step, and the NAME link edit control statement should reference the name specified by the GBLNAME parameter within that member.
4. Once modified, submit the ASMGBLS job to assemble and link-edit the link globals module. A new link globals module with the name specified by the GBLNAME parameter will be generated in the user load library identified in the ASMGBLS job.
5. Copy sample job LNKLNK8 (provided in the `AI Ivrs.SRCE` library) to a user source library.

6. Modify the sample job LNKLN18 to link the new link globals module you created along with the following INCLUDE statement:

```
INCLUDE ALALIB(EABXIMS)
```

where ALALIB references the `ALAvrs.LOAD` library.



Note: Instructions for modifying the sample job are described at the top of the job. Be sure to direct the output from the job to an appropriate user load library. The module resulting from this job is called ADALNI.

7. Place the newly linked ADALNI module in a load library available to the IMS MPP regions.

For general information regarding the installation of Adabas with IMS/TM and an explanation of the associated LGBLSET parameters, refer to the *Adabas for Mainframes* documentation > *Installing Adabas with TP Monitors*.

Adabas Auditing and Adabas Review

If you wish to install Adabas Auditing under a TP monitor which currently has Adabas Review installed, please observe the following points:

1. Your existing LGBLSET Review-related parameters (REVIEW=, REVHID=, RVCLNT=) should be maintained. This is in addition to the AUDIT=YES parameter required to install Adabas Auditing.



Note: The parameter values REVIEW=COR and/or RCVCLNT=COR are not currently compatible with AUDIT=YES therefore replace these parameter values with REVIEW=YES and/or RVCLNT=YES accordingly.

2. In your TP monitor Adabas link job which link-edits the Adabas link globals module with the Adabas Review TP monitor component, replace the Adabas Review component with the corresponding Adabas Auditing component according to the following table:

TP Monitor	Adabas Review Component	Adabas Auditing Component
Com-plete	RDBLXCOM	EABXCOM
CICS	RDBLXCIC	EABXCIC
Batch/TSO	RDBLXMVS	EABXMVS
IMS/TM	RDBLXIMS	EABXIMS

3. If you have installed REVUEX1 (Review User Field User Exit), then you will have previously linked your user exit to the appropriate Adabas Review TP monitor component using the sample member LREVUEX1 (provided in the `REVvrs.JOBS` library). Use this same sample member to

link your user exit to the equivalent Adabas Auditing TP monitor component. Refer to the table above for the names of the corresponding components.

8 Activating Auditing for LUW applications

- Enable LUW applications for auditing 48

The following section describes how to activate auditing for LUW applications.

Enable LUW applications for auditing

- Set the Natural Profile Parameter ADAPRM to on.
- Set the ACL variable WCLAUDIT=YES. This can be set as an environment variable, or in the wclclients.config file.



Note: Currently, for LUW, only Natural programs are eligible for auditing.