Adabas System Coordinator

Adabas System Coordinator Installation

Version 8.6.1

September 2025

ADABAS & NATURAL

Table of Contents

1 About this Documentation

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
`Monospace font`	Identifies service names and locations in the format *folder.subfolder.service*, APIs, Java classes, methods, properties.
Italic	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://documenta-tion.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://hub.docker.com/publishers/softwareag** 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 AG 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 Adabas System Coordinator Installation

⚠ **Important:** Before installing or upgrading, review the release notes, readmes, changes, system requirements, and installation or upgrade guide for the products you want to install. This documentation provides information you must know about the products before installing or upgrading, and also describes information you will need to provide during installation. Documentation is available at **https://documentation.softwareag.com/**.

This document describes how to install Adabas System Coordinator using installation jobs that are:

- generated by the Software GmbH's System Maintenance Aid (SMA), or
- taken from the job library on the installation medium and manually customized.

In either case, the relevant job numbers (prefixed by the Adabas System Coordinator product code COR) are the same and are referenced at the appropriate step of the installation procedure.

For information about using SMA, refer to the *System Maintenance Aid* documentation.

📄 **Note:** The Adabas System Coordinator installation medium contains several files. Always refer to the company's Product Delivery Report and Release Notes that accompany the medium for specific information that may modify the general installation procedures described here.

Installation Prerequisites
Before You Install
Installation Procedures
Verifying the Installation

3 Installation Prerequisites

This section describes the prerequisites for Adabas System Coordinator Version 8.6.

Operating Systems

Adabas System Coordinator is compatible with the z/OS operating system environment only.

> **Note:** For information regarding product compatibility with IBM platforms and any IBM requirements for Software GmbH products, please review the **Product Compatibility for IBM Platforms** web page.

> **Note:** Adabas System Coordinator currently executes in 31-bit addressing mode only.

Adabas

Adabas System Coordinator can be used with:

- Any supported level of Adabas, or
- Any supported level of Adabas Cluster Services, or
- Any supported level of Adabas Parallel Services.

Adabas Nucleus Memory Requirements

The use of any Adabas System Coordinator Add-on products will require memory in the affected Adabas nucleus jobs.

The minimum memory requirements can be approximated by first considering the Add-on product mix using the table below:

	COR	AAF	AFP	ATM	AVI
Memory requirement	+1 MB	+0.25 MB	+1 MB	+1 MB	+1 MB

In addition to this a further 15k is required for each defined Adabas thread (ADARUN NT=).

Example

The approximate memory requirement for an Adabas nucleus with ADARUN NT=85 running Adabas Fastpath is calculated as follows:

```
1 MB (for COR) + 1 MB (for AFP) + (15k * 85) = 3.25 MB ↵
```

Memory requirements will be greater if any of the AFPLOOK and AVILOOK database reporting tools are activated.

For AFPLOOK:

If activated using the following defaults:

- Maximum Files = 64
- Command/Descriptors per File = 32
- Maximum Concurrent Users = 100
- Maximum CIDs per User = 10

Then the additional nucleus memory required is approximately 100k.

If the Command/Descriptors per File and Maximum CIDs per User are kept at 32 and 10 respectively:

- Each additional file above 64 files will require 1k
- Each additional user above 100 users will require 288 bytes

For AVILOOK:

If activated AVILOOK will require an additional 200k of nucleus memory.

Natural

Natural is required by the Online Services application SYSCOR.

Any supported level of Natural can be used. Refer to the Natural documentation for more information.

If the Adabas System Coordinator client component CORS09 is in use at your site then refer to **Multi-Task Server Environments using CORS09** for additional prerequisite information.

Com-plete

Any supported level of Com-plete can be used. Refer to the Com-plete documentation for more information.

Adabas System Coordinator-based Add-on products

You can use any supported version of the following Adabas System Coordinator-based Add-on products in conjunction with Adabas System Coordinator version 8.6:

- Adabas SAF Security (AAF)
- Adabas Fastpath (AFP)
- Adabas Transaction Manager (ATM)
- Adabas Vista (AVI)

The following table describes the Add-on product maintenance required for supporting Adabas System Coordinator version 8.6:

Add-on Product	Version	Fix number
Adabas Fastpath	8.2 SP2	AFP822I005, AW822056, AW822057
	8.6 SP1	None required
Adabas SAF Security	8.4 SP1	None required
Adabas Transaction Manager	8.2 SP2	ATM822I003, AT822035
Adabas Vista	8.2 SP2	AVI822I006
	8.6 SP1	None required

> **Note:** For information regarding the necessary Adabas System Coordinator-based Add-on maintenance required for the version of Adabas you are running, refer to the section *Using COR-based Add-ons* in the release notes for the appropriate Adabas version.

Multi-Task Server Environments using CORS09

The Adabas System Coordinator client component CORS09 was made widely available in Version 8.2.2, Patch level 3, as a replacement for CORS07 for customers running Natural for zIIP in multi-task server environments.

For Adabas System Coordinator 8.3 SP1 and above, Natural zap NA97065 is a prerequisite for the continued use of CORS09 in conjunction with Natural for zIIP. The absence of NA97065 may result in a RSP101 Subcode 22.

4 Before You Install

This section describes actions which must be taken prior to performing Adabas System Coordinator installation.

License

The Adabas System Coordinator daemon requires an associated license file to make its services available.

> **Note:** If the Adabas System Coordinator license file is not available or erroneous, client jobs which require COR daemon services may not be able to initialize correctly and will receive RSP101 subcode 59s accompanied by message COR079E.

The Adabas System Coordinator license file can be transferred to the mainframe in the same way as the Adabas license and made available as a load module named CORLIC. Alternatively, the license file can be referred to by a "DDLCOR" DD statement in the COR daemon's job/started task. This is a fallback in case the CORLIC license load module cannot be loaded.

For installation information regarding the Adabas System Coordinator license file, refer to the section **Installation Procedure**.

For general information on Adabas licensing procedures, license check software and license files, refer to the *Mainframe Product Licensing* topic in the *Adabas for Mainframes* documentation.

Configuration File

Adabas System Coordinator operates correctly only if the configuration file is continuously available while the client is active. Operational procedures are necessary to ensure that the database where the configuration file (or the optional alternate configuration file) resides is active

- before any application opens to clients
- before any TP initialization processing that involves pseudo- or real database communication
- before any Coordinator daemons are started

Prior to beginning with the installation, allocate a database number and file number for the configuration file that is shared by Adabas System Coordinator, Adabas Fastpath, Adabas Vista, and Adabas Transaction Manager.

> **Notes:**

1. If an (optional) alternate configuration file is to be used, this must be allocated in a different database to the primary file.

2. It is your responsibility to ensure the alternate file has the same configuration content as the primary file. If either configuration file is modified, the equivalent change must also be made to the other configuration file. Failure to do so may lead to unpredictable results.

3. Both the primary and the alternate configuration files must be available at startup and shutdown of Coordinator daemons.

Adabas System Coordinator Daemon

Prior to beginning with the installation, a Node ID for each Adabas System Coordinator daemon must be allocated.

Using System Coordinator With Adabas Link Modules

Adabas System Coordinator is activated by linking an appropriate client component (CORS*nn*) with the LNKGBLS module. The LNKGBLS module must be re-assembled, specifying the parameter `COR=YES` in the LGBLSET macro. The Coordinator will not activate if the component is incorrectly linked.

> **Note:** Adabas client-based products are not compatible with the Adabas DBID/SVC routing feature.

Use of Unmodified ADALNK

The Coordinator client component is activated by binding a stub module to the Client Adabas Link Module (ADALNK or other). This stub module is for use in client environments only. In previous versions it has been a documented restriction that the ADALNK module used by the COR daemon and Adabas servers must not contain the COR client stub. *This remains the recommended procedure.* However, in this version COR will auto-detect and bypass invalid client stub invocation in the COR daemon and Adabas, Entire System Server, or Adabas Review Hub nuclei.

You must still ensure that you use an unmodified ADALNK in Adabas utility and Entire Net-Work jobs.

Use of Client-Side ADALNK User Exits with Adabas System Coordinator

Your site may attach user exits to the Adabas Link module such as LUEXIT1/UEXITB and LUEXIT2/UEXITA. These exits will see Adabas command traffic in a form that is (mostly) unaffected by products such as Adabas Vista, Fastpath, etc. However, some sites have a need for exits to see Adabas command traffic in its modified form. If your site needs to see this then you can use some special purpose exits to achieve it:

■ IEXIT1 receives control after a command has been adjusted by products such as Vista but before the command is passed through the Adabas router.

■ IEXIT2 receives control after a command has been completely processed through the Adabas router and before the command result is processed by the after-processing of products such as Vista.

These exits must use CSECT names IEXIT1 and IEXIT2. They receive control in 31-bit addressing mode, they must be re-entrant and should specify AMODE 31, RMODE ANY.

The exits are linked with the COR stub and the link module by adding link-edit INCLUDE statements to the relevant COR link job. In addition to linking the exits a specific step is needed to activate for each client job (so you are able to choose which jobs they are used with and which ones they are not). In the System Coordinator Runtime Controls panel set Use additional exits to Y. See the section Maintain Client Runtime Controls in the Online Services documentation for further information.

At entry to the exit(s), the registers contain the following:

Register	Contents
1	Address of the UB.
2	Address of an 18-word save area (CICS environments)
13	Address of an 18-word save area (non-CICS environments)
14	Return address
15	Entry point address: IEXIT1 or IEXIT2

Any registers except register 15 that are modified by the user exits must be saved and restored. On return from IEXIT1/2 register 15 must be set to zero.

> **Note:** IEXIT1/2 can be mixed with LUEXIT1/UEXITB, LUEXIT2/UEXITA.

5 Installation Procedures

The Installation Medium

Review the *Software GmbH Product Delivery Report* that accompanies the release package before restoring the release data to disk. Information in this report supersedes the information in this documentation.

The installation medium contains the following data sets in the sequence indicated in the report:

Data Set	Contents
COR*vrs*.LOAD	COR load modules
COR*vrs*.INPL	SYSCOR INPL file
COR*vrs*.ERRN	SYSCOR error messages file
COR*vrs*.SRCE	Source modules
COR*vrs*.JOBS	Installation jobs
COR*vrs*.SYSF	Base configuration file

where *vrs* in data set names represents the version, revision, and system maintenance level of the product.

Installation Overview

The steps needed for a successful installation are as follows:

Step	Description	Required	Job Name
0	Check, Prepare, and Install the Product License File	Yes	
1	Restore the libraries from the installation medium	Yes	
2	Load (INPL) the SYSCOR application	Yes	CORI061
3	Load the configuration file and prepare SYSCOR	Yes	CORI050
4	Assemble the configuration module	Yes	CORI055
5	Add the System Coordinator to the Adabas clients	Yes	CORI060, CORI080x
6	Define the System Coordinator group and daemons	Required if a COR daemon is to be used	
7	Install the CICS node error program (optional)	Optional	
8	Create startup procedures for the Adabas System Coordinator daemons	Required if a COR daemon is to be used	

Step	Description	Required	Job Name
9	Create System Coordinator daemon disk files	Optional, depending on which products are installed and how they are configured.	CORI040
10	Define runtime controls for Client jobs and TP systems	Optional	

System Programming Considerations

In a multi-systems environment a Coordinator daemon is normally defined for each system. In a parallel sysplex daemons use the IBM XCF facility to communicate. All COR daemons are defined (in the COR configuration file) with the same group name, and this name is used as the XCF group name. The group name selected must be unique to the COR daemon group, and must *not* be the same as the group name (CLUGROUPNAME) selected for any Adabas Cluster Services database.

The Adabas System Coordinator daemon must execute:

- from an authorized load library; and

- at a higher priority than the TP monitors, databases, and jobs it is used to coordinate.

Installation Procedure Steps

Following is the general Adabas System Coordinator installation procedure. The actual installation depends on your particular requirements and the specific contents of the release package provided by Software GmbH for your site. Information in the release package is intended for your system. If that information differs from the information in this section, use the release package information or contact technical support for assistance.

- Step 0: Check, Prepare, and Install the Product License File
- Step 1: Copying the Medium Contents to Disk
- Step 2: Load (INPL) the SYSCOR Application (Job I061)
- Step 3: Load the Configuration File and Prepare SYSCOR (Job I050)
- Step 4: Assemble the Configuration Module (Job I055)
- Step 5: Add the System Coordinator to the Adabas Client (Jobs I060, I080x)
- Step 6: Define a System Coordinator Group
- Step 7: Install the CICS Node Error Program (Optional)
- Step 8: Create Startup Procedures for the System Coordinator Daemon(s)
- Step 9: Create System Coordinator daemon disk files

- Step 10: Define Runtime Controls for Client Jobs and TP Systems

Step 0: Check, Prepare, and Install the Product License File

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

> **Note:** Forty days before the license expires, license check failure messages will be generated. Adabas System Coordinator will continue to function, but these messages serve as a warning that it is time to obtain a new license.

In this step, you will prepare the license file (obtain it from email or the installation tape and store it on your z/OS system), 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 email attachment. Before you can install the license, you must transfer it from email or the installation tape and store it on your z/OS system. This section describes how to do this for a license distributed either by email or on the installation tape.

To prepare the license file from an email attachment, complete the following steps:

1. Transfer the license to z/OS, as described in the section *Transferring a License File from PC to a z/OS Host Using FTP* within the *Software GmbH Mainframe Product Licensing* topic in the *Adabas for Mainframes* documentation.

2. Verify that the transferred license file is stored in a source library (with `RECFM=F` or `FB` and `LRECL=80`), taking care to preserve its format as ASCII.

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

- Verify that the license file has been stored from the tape into a source library (with `RECFM=F` or `FB` and `LRECL=80`), taking care 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 file into a load module (CORLIC), which is then loaded by the appropriate job/started task.

- Reference the license file in the appropriate job/started task using a DD statement.

This section describes both methods.

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

1. Review and modify the sample job ASMLICAM, supplied in the ADA*vrs*.JOBS library, as follows:

   - Ensure the STEPLIB DD statement resolves to the license load library (MLC*vrs*.LOAD).

   - Ensure the SYSUT1 DD statement resolves to the data set containing the Adabas System Coordinator license file that you transferred to z/OS earlier.

   - Ensure the generated load module is named CORLIC.

   - Ensure the SYSLMOD DD statement resolves to an appropriate user load library.

     > **Note:** This user load library must also be included in the STEPLIB concatenation for the appropriate job/started task.

2. Submit the modified ASMLICAM. This job runs the MAKE function of the LICUTIL utility to convert the license text file into an assembler source module. ASMLICAM then assembles and links the assembler source to generate a load module called CORLIC, which is stored in the specified user load library. For more information about the LICUTIL utility, refer to the section *Using The License Utility: LICUTIL* within the *Software GmbH Mainframe Product Licensing* topic in the *Adabas for Mainframes* documentation.

3. Update your job/started task to reference the user load library so that CORLIC will be loaded, as described in **Step 8: Create Startup Procedures for the System Coordinator Daemon(s)**.

To reference the license file by DDLCOR statement, complete the following steps:

1. Ensure that any previously created CORLIC license load module is inaccessible to the job/started task (loading the CORLIC license load module is attempted first, and only if this is unsuccessful is a read from the DDLCOR data set attempted).

2. Update your job/started task as described in **Step 8: Create Startup Procedures for the System Coordinator Daemon(s)**.

Step 1: Copying the Medium Contents to Disk

If you are using System Maintenance Aid (SMA), refer to the SMA documentation (included on the current edition of the Natural documentation CD). If you are not using SMA, perform steps 1a, 1b and 1c as described in this section:

- Step 1a: Copy Data Set COPY.JOB from Medium to Disk
- Step 1b: Modify COPY.JOB
- Step 1c: Submit COPY.JOB

> **Note:** If the data sets for more than one product are delivered on the medium, the data set COPY.JOB contains the JCL to unload the data sets for all delivered products from the medium to your disk. After that, you will have to perform the individual install procedure for each component.

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

The data set COPY.JOB (label 2) contains the JCL to unload all other existing data sets from medium to disk. To unload COPY.JOB, use the following sample JCL:

```
//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=<Tnnnnn>),
// LABEL=(2,SL)
//SYSUT2 DD DSN=<hilev>.COPY.JOB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=3390,VOL=SER=<vvvvvv>,
// SPACE=(TRK,(1,1),RLSE),
// DCB=*.SYSUT1
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//
```

where:

```
<hilev>   is a valid high level qualifier
<Tnnnnn> is the tape number
<vvvvvv> is the desired volser
```

Step 1b: Modify COPY.JOB

Modify the COPY.JOB to conform with your local naming conventions and set the disk space parameters before submitting this job:

- set HILEV to a valid high level qualifier
- set LOCATION to a storage location
- set EXPDT to a valid expiration date

Step 1c: Submit COPY.JOB

Submit COPY.JOB to unload all other data sets from the medium to your disk.

Step 2: Load (INPL) the SYSCOR Application (Job I061)

Use sample job CORI061 to load the SYSCOR online administration and error messages file into Natural.

Step 3: Load the Configuration File and Prepare SYSCOR (Job I050)

System Coordinator and related products operate according to definitions contained in the configuration file. You must allocate a new Adabas file for the Version 8.3 configuration file and load COR*vrl*.SYSF into it using the supplied sample job CORI050. The online services will guide you through the steps required to make the new file ready for use, including conversion from previous versions of System Coordinator.

≫ **To load the configuration file**

1   Load the Adabas System Coordinator configuration file from the distribution medium using the standard Adabas load utility ADALOD. Use sample job CORI050. If you are using an alternate configuration file you need to run this job to initialize both files.

2   If Natural Security is installed, define the libraries SYSCOR and SYSMP*vrs* (where *vrs* is the version you are installing, for example 821) and protect as required. You may define MENU as the startup transaction for SYSCOR. DO NOT define a startup transaction for SYSMP*vrs*.

3   Use the following parameter to define the Natural session where SYSCOR is to be used:

```
LFILE=(152,dbid,fnr<,passw><,ciph>)
```

where *dbid* and *fnr* define the primary Adabas System Coordinator file.

Alternatively, assemble the Natural parameter module with:

```
NTFILE ,ID=152,DBID=dbid,FNR=fnr
```

≫ To convert a previous version's definitions to version 8 format

1   Logon to library SYSCOR and enter `MENU`. SYSCOR will detect that the configuration file is new and will guide you through the steps required to copy and convert the definitions from a previous version.

2   Repeat this procedure for each configuration file to be converted. The procedure only needs to be done once for each configuration file, regardless of how many products use it. Other products may have additional conversion requirements.

Step 4: Assemble the Configuration Module (Job I055)

Adabas System Coordinator parameters are located in the configuration file. At job start, the Adabas System Coordinator needs to know the location of this file. This information is kept in the configuration module.

Create the configuration module by assembling the CORMCFG parameters defining the SVC, database, and file number of the Adabas System Coordinator configuration file.

Keyword	Description
SVC=	Your installation's Adabas SVC number
DBID=	Database number for the System Coordinator configuration file
FNR=	File number for the System Coordinator configuration file
ADBID=	Database number for the alternate System Coordinator configuration file. If not specified, an alternate file will not be used.
AFNR=	File number for the alternate System Coordinator configuration file. If not specified, an alternate file will not be used.
SF148=WAIT	Use this keyword if you want client jobs to wait when the specified configuration file is not active. If you omit this keyword, the `RETRY=` setting takes effect (see below).
CRITICAL=	Use this keyword if you want System Coordinator to check for availability and correct functioning of supported add-on products. You may specify one or more of the following, separated by commas: ■ AVI - AdabasVista ■ AFP - Adabas Fastpath ■ ATM - Adabas Transaction Manager If any critical product is not functioning correctly, all Adabas requests will be rejected with response code 101, subcode 59.
DMWAIT=	Specifies a maximum time (in minutes) that the System Coordinator daemon will wait for the configuration file database to be activated. If not specified, the daemon will wait indefinitely. The default is 60.
ZOSDUMP=	This option applies to z/OS installations only and determines the style of dump the System Coordinator daemon produces, in the event of an abnormal termination.

Keyword	Description
	The default, ABEND, produces a standard operating system dump to SYSUDUMP, SYSABEND or SYSMDUMP. SVC produces an SVC dump on the system dump datasets. This can often be quicker than producing a standard dump. The daemon issues message: COR037I SVC dump created and the operating system writes messages to the system log to identify the dump dataset. If the SVC dump fails for any reason, the daemon issues message: COR038E SVC dump failed, R15: hhhhhhhh and a standard dump is produced.
RETRY=	RETRY specifies the frequency (number of Adabas calls issued in this client job) at which System Coordinator in the client job will retry access to its configuration file after a response 148. You must specify RETRY in the numeric range from 100 to 1,000,000. The default is 1000. Client services provided by System Coordinator and its supported add-on products are not available until the configuration file becomes active. RETRY only takes effect if you have not specified `SF148=WAIT`.
SCERR=	Specifies the action to be taken if there is an error when initializing System Coordinator in the client job. The possible values are CONT for continue or WAIT.
MCASE=	Specifies the case used for messages. The possible values are M for Mixed Case or U for Upper Case.
STARTMSG=	Specifies an optional message up to 80 characters at successful initialization of System Coordinator in the client job. This message may be conditional on one or more products. See MSGPROD below.
MSGPROD=	Used in conjunction with STARTMSG. Outputs a message when a product initializes. Possible values are AFP for Adabas Fastpath, ATM for Adabas Transaction Manager, or AVI for Adabas Vista.

Name the resulting load module CORCFG (this is required).

Use sample job CORI055.

The following parameters are no longer supported and are documented here for reference only:

Keyword	Description
DEMO=	No longer supported.
AFPPARM=	No longer supported.
ATMPARM=	No longer supported.
AVIPARM=	No longer supported.
SAFPARM=	No longer supported.

Step 5: Add the System Coordinator to the Adabas Client (Jobs I060, I080x)

The client components are named CORS0*n* where n is a subsystem suffix.

First, you must re-assemble the Adabas LNKGBLS table, specifying the parameter `COR=YES` in the LGBLSET macro.

Next, link the appropriate Adabas System Coordinator client component with your Adabas link components (note - the resulting COR-enabled Adabas link components must only be used for client jobs).

Finally, to enable the Adabas System Coordinator client process, make the following components available in STEPLIB (or COMPLIB or DFHRPL if indicated in the table below):

- the COR-enabled Adabas link components;
- the generated configuration module CORCFG; and
- the Adabas System Coordinator load library.

Important Note:

All Adabas System Coordinator components involved in the client process must be of the same version, release, sm-level, and patch-level. For example, for a particular client process, the version of the CORS0*n* component included in the COR-enabled Adabas link module must be identical to the version of the Adabas System Coordinator library made available to that client process.

For ease of reference, the table below lists a number of job types, the corresponding CORS0*n* component, the corresponding sample job, and any relevant installation notes that may apply.

Table Notes:

1. You must use the linkage editor options specified in the sample jobs

2. The sample install jobs referenced below may not match what is needed for the release of Adabas used at your site. Please refer to your Adabas installation documentation and JCL to ensure you use the appropriate JCL for your release, adjusted for the needs of the sample jobs supplied here.

Job Type	CORS0*n* Component	Sample Job and Installation notes
Batch/TSO	CORS01	CORI060
Com-plete	CORS02	CORI080C **Note:** 1. The COR-enabled Adabas link components should be available in COMPLIB.

Job Type	CORS0*n* Component	Sample Job and Installation notes
		2. The generated CORCFG module should be available in COMPLIB.
		3. The COR modules should be available in COMPLIB.
CICS Command Level	CORS03	CORI080B **Note:** 1. The COR-enabled Adabas link components should be available in DFHRPL. 2. The generated CORCFG module should be available in DFHRPL. 3. It is recommended that the COR modules should be available in DFHRPL. We do not recommend loading from STEPLIB as it can interfere with correct CICS threadsafe operation for Adabas. 4. If you are not running threadsafe and still decide to use STEPLIB you can only do so for COR modules other than CORCFG and CORKRN – these two must be present in the RPLLIB. 5. If you are not using the CICS program autoinstall feature, you will need to define the Coordinator Client modules (CORKRN, CASPXY, CASKRN), and the configuration module (CORCFG) to CICS. All of the modules should be defined with the following characteristics: Language: Assembler ; RELoad :No ; DAtalocation: Any ; EXECKey : User. For sites that require CICS definitions the member CORI080R in the Coordinator JOBS library shows the input needed for all modules (for all sibling products too). **Note:** For CICS, ensure that the `LUSAVE` parameter in the Adabas link module is set to at least 72. We recommend that you also use the `XWAIT=YES` parameter. **Note:** For CICS, System Coordinator uses Main Temporary Storage queues to store important control information. By default the queue names begin "COR" followed by a hexadecimal "FF" – you can change this prefix using the *Client Runtime Latency Controls*. The queue name includes the CICS job-name to ensure that it is unique in a multi-job CICS system. However, the CSD TSMODEL definition for these temporary storage queues should not specify a POOLNAME or REMOTESYSTEM, as they cannot be shared.

Job Type	CORS0*n* Component	Sample Job and Installation notes
IMS	CORS05	CORI080G
Multi-task Batch (for Natural RPC Server, Natural Development Server, Adabas SOA Gateway, Adabas SQL Gateway)	CORS07 / CORS09	CORI080H **Note:** See section **Installing Adabas System Coordinator in multi-task batch environments**. **Note:** 1. CORS07 is for sites which **do not use** Natural for zIIP. 2. CORS09 is for sites which **do use** Natural for zIIP.
Other 3rd party or in-house multi-task server environments.	CORS07 / CORS09	CORI080H **Note:** See section **Installing Adabas System Coordinator in 3rd party or in-house multi-task server environments**. **Note:** 1. CORS07 is for sites which **do not use** Natural for zIIP. 2. CORS09 is for sites which **do use** Natural for zIIP.
NIM: Natural TP monitor	CORS07	CORI080N **Note:** See section *Installing Adabas System Coordinator with NIM* for further information.
Triggers and Stored Procedures **Note:** Define a System Coordinator runtime control of type "SPATs". The SPATs option is on the second screen of selectable runtime control types, accessible by selecting "more choices for type or" when adding the runtime control.	CORS08	CORI080S **Note:** You must ensure that this ADALNK is used **only** by Adabas nuclei running Triggers and Stored Procedures and is not used in any client job or Coordinator daemon.

Step 6: Define a System Coordinator Group

Define your System Coordinator group and member(s). This is required if you intend to use:

- Adabas Fastpath

- clustered applications with dynamic transaction routing.

Job parameters for each product also contain settings that are relevant to the operation of the Adabas System Coordinator. For more information, see SYSCOR Administration.

Step 7: Install the CICS Node Error Program (Optional)

The node error program CORNEP is used by sites running CICS command-level applications in CICS/ESA or CICS Transaction Server for z/OS. It is not an essential component, but it does improve efficiency when reclaiming user memory after user sessions terminate.

CORNEP must be called as a started task (with Transaction ID ANEP) from the real CICS node error program DFHZNEP. If you do not use DFHZNEP, a sample is provided on the source library. If you do use DFHZNEP, you will need to implement the code for starting CORNEP into your own DFHZNEP as shown in the provided sample source.

Following are the required CICS resource definition parameters for CORNEP:

```
Language: Assembler
RESident: No
Datalocation: Any
EXECKey: User
```

> **Note:** To use CORNEP, assemble your Adabas link module with `PARMTYP=ALL` on the ADAGSET (Version 7 link module) or LGBLSET (Version 8 link module) macro.

> **Note:** CORNEP must be called only from DFHZNEP.

Step 8: Create Startup Procedures for the System Coordinator Daemon(s)

The following is a job example for running an Adabas System Coordinator daemon:

```
//SYSCO1 PROC
//*--------------------------------------------------------------*
//* System Coordinator SYSCO v.r.s. STARTUP *
//*--------------------------------------------------------------*
//SYSCO1 EXEC PGM=SYSCO,REGION=0M,TIME=1440
//STEPLIB DD DISP=SHR,DSN=SAG.CORvrs.LOAD
//        DD DISP=SHR,DSN=SAG.MLCvrs.LOAD
//        DD DISP=SHR,DSN=SAG.ADAvrs.LOAD
//SYSUDUMP DD SYSOUT=*
```

```
//CORDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//DDPRINT DD SYSOUT=*
//DDCARD DD DISP=SHR,DSN=SAG.CORvrs.SRCE(CORCNTL)
//DDLCOR DD DISP=SHR,DSN=SAG.CORvrs.LICENSE
//*
```

The file referenced by the DDCARD statement should contain the following control statements:

```
PRODUCT=CAS
PRODUCT=AFP (If FASTABM is to be run)
FORCE=NO
```

Licensing requires the following:

1. Modules LICMAIN and LICUTIL are available to the job/started task at start up. These modules are distributed in the MLC*vrs*.LOAD library. You must either copy LICMAIN and LICUTIL into a library that is already in the STEPLIB concatenation or place them in an appropriate user library and include that library in the STEPLIB concatenation.

2. Verification that the Adabas System Coordinator license file is available to the job/started task at start up. Do either of the following:

   ■ Verify that the CORLIC license load module installed in **Step 0: Check, Prepare, and Install the Product License File** is available to the job/started task by either copying it into a library that is already in the STEPLIB concatenation, or place it in an appropriate user library and include that library in the STEPLIB concatenation.

   ■ If the license file is to be referenced by DDLCOR statement (as shown in the sample JCL above), verify that there is no CORLIC license load module available to the job/started task and that the following DD statement is defined instead:

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

   where *dsn* is the dataset name of the license file loaded from the tape or provided by email (in ASCII format). Note that *dsn* could reference a member in a partitioned data set.

Step 9: Create System Coordinator daemon disk files

Some products require a daemon disk file. This is described in the relevant product documentation. Tailor and run job CORI040 to create each required disk file. Examples of products for which you might need to create disk files are:

■ System Coordinator's latency file for latency management of TP systems.

■ Adabas Transaction Manager's recovery file.

Step 10: Define Runtime Controls for Client Jobs and TP Systems

System Coordinator can be installed for all client jobs, but will be inactive until runtime controls are defined. Controls are defined in the SYSCOR Natural application, using the Maintenance menu. Refer to the *Online Services* section for further information.

Alternatively, you can delay this task until you have installed the appropriate add-on product(s). You may then use any of the supplied maintenance applications (SYSAFP, SYSAVI or SYSATM).

Installing Adabas System Coordinator in multi-task batch environments

Multi-task batch environments have special requirements. These are discussed below for the most common environment types.

- Installing the Adabas System Coordinator with the Adabas SQL Gateway in z/OS
- Installing Adabas System Coordinator with Natural RPC Batch Server
- Installing Adabas System Coordinator with Natural Development Server (NDV)
- Installing Adabas System Coordinator with the Adabas SOA Gateway
- Installing Adabas System Coordinator in 3rd party or in-house multi-task server environments
- Installing Adabas System Coordinator with NIM

Installing the Adabas System Coordinator with the Adabas SQL Gateway in z/OS

- Background
- Common Installation Steps
- Basic Multi-Thread (not APF authorized) Install
- Authorized Multi-Thread Install

Background

The Adabas SQL Gateway uses z/OS UNIX Systems Services to run as a multi-threaded server. This requires re-entrant operation which has additional installation considerations for products based upon Adabas System Coordinator such as Adabas Fastpath, Adabas Vista and Adabas Transaction Manager.

The Adabas SQL Gateway uses z/OS UNIX Systems Services to run either in basic or authorized multi-threaded server mode. This choice affects the method of installation of the Adabas System Coordinator as the base for Adabas Fastpath, Adabas Vista, Adabas Transaction Manager, etc.

Specifically, it is important that a single copy of the Adabas Link Module (ADALNKR) containing the System Coordinator interface (CORS07) is used, and that this module is linked as specified in the sample CORI080H job.

There are two ways to run the Adabas SQL Gateway which directly affect the way Adabas System Coordinator must be installed:

- Basic multi-thread (not APF authorized): When the Adabas SQL Gateway runs without APF authorization it does not activate its low-level RACF interface.

- Authorized multi-thread (APF authorized): In its authorized state the Adabas SQL Gateway activates its RACF interface.

Common Installation Steps

≫ **The following steps are required for both installation types:**

1   Complete normal SQL Gateway installation (and make sure it works correctly in isolation) before adding the System Coordinator interface. (See the *Installation for z/OS* section of the Adabas SQL gateway documentation).

2   Assemble the Adabas Link routine globals module LNKGBLS and link ADALNKR using sample job CORI080H. This COR-enabled ADALNKR should be made available in the appropriate STEPLIB concatenation.

3   Modify the Server Job CNXADA. Include the Coordinator load library, plus any installed add-on product libraries (Fastpath, Vista, ATM), in the STEPLIB concatenation. For the authorized installation, all libraries must be APF-authorized.

4   Define the SQL Gateway started task as job type multi-TCB to the desired product(s) in the appropriate administration center: Adabas System Coordinator (SYSCOR), Adabas Fastpath (SYSAFP), Adabas Vista (SYSAVI) and/or Adabas Transaction Manager (SYSATM). See the relevant product documentation for details. The Multi-TCB option is on the second screen of selectable runtime control types, accessible by selecting "more choices for type or" when adding the runtime control.

Basic Multi-Thread (not APF authorized) Install

≫ **for this type of operation:**

1   Ensure that SQL Gateway basic multi-thread installation is completed before adding the System Coordinator interface.

2   Re-link the module named CNXRUNA with the REUS attribute, replacing ADALNK with ADAUSER. Here is a sample of the required JCL:

```
//LKED    EXEC PGM=IEWL,PARM='REUS'
//ADALIB   DD DSN=ADABAS.VVRS.LOAD,DISP=SHR    ... Adabas library ...
//SYSLMOD  DD DSN=PPP.CONNX.LOAD,DISP=SHR      ... Connx Library ...
//SYSPRINT DD SYSOUT=*
//SYSLIN   DD *
 REPLACE ADABAS   ... Delete ADALNK entry ...                              ↵

 INCLUDE SYSLMOD(CNXRUNA)
 INCLUDE ADALIB(ADAUSER)
 ENTRY MAIN
 NAME CNXRUNA(R)
/*
```

The Linkage Editor will produce warning messages indicating that some Csects are not re-usable. These can be ignored.

3   Re-link the module named CNXADA0B with the REUS attribute, replacing ADALNK with ADAUSER. Here is a sample of the required JCL:

```
//LKED    EXEC PGM=IEWL,PARM='REUS'
//ADALIB   DD DSN=ADABAS.VVRS.LOAD,DISP=SHR    ... Adabas library ...
//SYSLMOD  DD DSN=PPP.CONNX.LOAD,DISP=SHR      ... Connx Library ...
//SYSPRINT DD SYSOUT=*
//SYSLIN   DD *
 REPLACE ADABAS       ... Delete ADALNK entry ...                         ↵

 INCLUDE SYSLMOD(CNXADAOB)
 INCLUDE ADALIB(ADAUSER)
 ENTRY MAIN
 NAME CNXADAOB(R)
/*
```

The Linkage Editor will produce warning messages indicating that some Csects are not re-usable. These can be ignored.

Do *not* specify the RENT atrribute for either of these modules. The linked modules must be re-usable but not re-entrant. Failure to do this will result in the System Coordinator failing with S0C4 (addressing) abends.

4   Ensure that the ADARUN parameter `PROG=RENTUSER` is specified in order that the COR-enabled ADALNKR (created by CORI080H and made available in the STEPLIB concatenation) is used.

Authorized Multi-Thread Install

≫ **for this type of operation:**

1   Ensure that SQL Gateway authorized multi-thread installation is completed before adding the System Coordinator interface.

2   Re-link the module named CNXRUNA with the RENT and AC=1 attributes, replacing ADALNK with CORLNKR. Here is a sample of the required JCL:

```
//LKED    EXEC PGM=IEWL,PARM='RENT,AC=1'
//CORLIB   DD DSN=SAG.CORVRS.LOAD,DISP=SHR   ... COR library ...
//SYSLMOD  DD   DSN=PPP.CONNX.LOAD,DISP=SHR  ... Connx Library ...
//SYSPRINT DD   SYSOUT=*
//SYSLIN   DD   *
 REPLACE ADABAS   ... Delete ADALNK entry ...                              ↵

 INCLUDE SYSLMOD(CNXRUNA)
 INCLUDE CORLIB(CORLNKR)
 ENTRY MAIN
 NAME CNXRUNA(R)
/*
```

3   Re-link the module named CNXADA0B with the RENT and AC=1 attributes, replacing ADALNK with CORLNKR. Here is a sample of the required JCL:

```
//LKED    EXEC PGM=IEWL,PARM='RENT,AC=1'
//CORLIB   DD DSN=SAG.CORVRS.LOAD,DISP=SHR   ... COR library ...
//SYSLMOD  DD   DSN=PPP.CONNX.LOAD,DISP=SHR   ... Connx Library ...
//SYSPRINT DD   SYSOUT=*
//SYSLIN   DD   *
 REPLACE ADABAS   ... Delete ADALNK entry ...                              ↵

 INCLUDE SYSLMOD(CNXADAOB)
 INCLUDE CORLIB(CORLNKR)
 ENTRY MAIN
 NAME CNXADAOB(R)
/*
```

Installing Adabas System Coordinator with Natural RPC Batch Server

In order for execution with a Natural RPC Batch Server using multiple tasks you must do the following:

1. A *multi*-TCB type runtime control must be defined for the Natural RPC Batch Server job. Multi-TCB type option is on the second screen of selectable runtime control types…select *more choices for type* when adding the runtime control to see the *multi*-TCB choice.

2. Link the Adabas System Coordinator client component (CORS09 if Natural for zIIP is in use or CORS07 if it is not) to the re-entrant Adabas link module using sample job CORI080H. The resulting COR-enabled link module must be named according to the requirements of the Natural RPC Batch Server (for example, ADALNKR) and must be linked with the attributes specified in the sample job.

3. Use the Natural profile parameter `ADANAME` and set `ADANAME=ADALNKR` (or whatever you have named the COR-enabled link module). This will cause Natural to load ADALNKR dynamically at runtime.

   Instead of using `ADANAME` you may link the COR-enabled link module with the Natural RPC Server front-end, however this will only operate correctly if the Natural front-end is loaded from non-authorized libraries, otherwise protection exceptions may occur. Our recommendation is therefore to use `ADANAME`.

4. The COR load library must be added to the STEPLIB concatenation of the Natural RPC Batch Server job.

5. The load library for the additional products being used (Fastpath, Vista, Transaction Manager etc) must be added to the STEPLIB concatenation for the Natural RPC Batch Server job.

6. Restart the Natural RPC Batch Server.

Installing Adabas System Coordinator with Natural Development Server (NDV)

- Installation Steps
- Session Timeout when using the Natural Development Server

Installation Steps

In order for execution with a Natural Development Server using multiple tasks you must do the following:

1. A *multi*-TCB type runtime control must be defined for the Natural Development Server job. Multi-TCB type option is on the second screen of selectable runtime control types…select *more choices for type* when adding the runtime control to see the *multi*-TCB choice.

2. Link the Adabas System Coordinator client component (CORS09 if Natural for zIIP is in use or CORS07 if it is not) to the re-entrant Adabas link module using sample job CORI080H. The resulting load module must be named according to the requirements of the Natural Development

Server (for example, ADALNKR) and linked with the attributes specified in sample job CORI080H.

3. Use the Natural profile parameter `ADANAME` and set `ADANAME=ADALNKR` (or whatever you have named the COR-enabled link module). This will cause Natural to load ADALNKR dynamically at runtime. Instead of using `ADANAME` you may link the COR-enabled link module with the Natural Development Server front-end, however this will only operate correctly if the Natural front-end is loaded from non-authorized libraries, otherwise protection exceptions may occur. Our recommendation is therefore to use `ADANAME`.

4. The COR load library must be added to the STEPLIB concatenation of the Natural Development Server job.

5. The load library for the additional products being used (Adabas Fastpath, Adabas Vista, Adabas Transaction Manager etc) must be added to the STEPLIB concatenation for the Natural Development Server job.

6. Restart the Natural Development Server.

Session Timeout when using the Natural Development Server

To ensure the session timeout is performed under the control of the Natural Development Server, the implementation of the following session timeout rules is recommended when running a zIIP-enabled Natural Development Server and Adabas System Coordinator client component CORS09 linked to the reentrant Adabas link module:

1. **Natural Development Server configuration parameter SESSION_TIMEOUT**

   Use the following format for the `SESSION_TIMEOUT` configuration parameter:

   ```
   m <numeric value greater than 0>,n <numeric value greater than 0>
   ```

   where m and n are in minutes. For more information on this parameter, see the NDV Configuration Parameters section in the Natural Development Server documentation.

2. **Adabas System Coordinator Client Runtime Controls**

   ▪ Maximum idle time (sec)

   Ensure the Adabas System Coordinator client runtime control `Maximum idle time (sec)..` is set to an equivalent value greater than or equal to the value ($m + n$), where m and n are the values specified for the Natural Development Server configuration parameter `SESSION_TIMEOUT`.

   For example, if the NDV configuration parameter `SESSION_TIMEOUT` is:

```
SESSION_TIMEOUT = 30,120
```

then (m + n) = 30 + 120 = 150 minutes = 9000 seconds. So the recommended setting for Adabas System Coordinator client runtime control 'Maximum idle time' is:

```
Maximum idle time (sec)..: 9000
```

■ Generate RSP009/79 (Y/N)

With session timeout under the control of the Natural Development server, you must set the Adabas System Coordinator client runtime control Generate RSP009/79(Y/N) to N as follows:

```
Generate RSP009/79 (Y/N).: N (until 0_____ seconds elapse)
```

Installing Adabas System Coordinator with the Adabas SOA Gateway

In order for execution with the Adabas SOA Gateway you must do the following:

1. A *multi-TCB* type runtime control must be defined for the Adabas SOA Gateway job. Multi-TCB type option is on the second screen of selectable runtime control types…select *more choices for type* when adding the runtime control to see the *multi-TCB* choice.

2. Link the Adabas System Coordinator client component (CORS07) to the re-entrant Adabas link module using sample job CORI080H. The resulting load module must be named according to the requirements of the Adabas SOA Gateway (for example, ADALNKR) and linked with the attributes specified in sample job CORI080H.

3. The resulting load module must be named according to the requirements of the Adabas SOA Gateway (for example, ADALNKR) and linked with the attributes specified in sample job CORI080H.

4. The COR load library must be added to the STEPLIB concatenation of the Adabas SOA Gateway job.

5. The load library for the additional products being used (Fastpath, Vista, Transaction Manager etc) must be added to the STEPLIB concatenation for the Adabas SOA Gateway job.

6. Restart the Adabas SOA Gateway.

Installing Adabas System Coordinator in 3rd party or in-house multi-task server environments

This section applies to 3rd party and customer-written multi-task batch applications designed to use the re-entrant Adabas link module.

> **Note:** For obvious reasons, the information here is only a guideline on what may be appropriate. The specifics of what is actually appropriate are dictated by the design of the server in question. We recommend for you to check with support before implementing.

In order for execution with 3rd party or in-house multi-task server environments you must do the following:

1. A *multi-TCB* type runtime control must be defined for the multi-task server job. Multi-TCB type option is on the second screen of selectable runtime control types…select *more choices for type* when adding the runtime control to see the *multi-TCB* choice.

2. Link the Adabas System Coordinator client component (CORS07) to the re-entrant Adabas link module using sample job CORI080H. The resulting load module must be named according to the requirements of the multi-task server (for example, ADALNKR) and linked with the attributes specified in sample job CORI080H.

3. The resulting load module must be named according to the requirements of the multi-task server (for example, ADALNKR) and linked with the attributes specified in sample job CORI080H.

4. The COR load library must be added to the STEPLIB concatenation of the multi-task server job.

5. The load library for the additional products being used (Fastpath, Vista, Transaction Manager etc) must be added to the STEPLIB concatenation for the multi-task server job.

6. Restart the multi-task server.

Usually, the server application must ensure that either:

■ the task which causes the COR environment to initialize (usually the first task to issue an Adabas call) stays active for the duration of the job

or

■ the load modules required for the product are pre-loaded by a task which stays active for the duration of the job. The modules required for COR are listed above (under Job Type CICS Command Level) and each product names its required modules.

Installing Adabas System Coordinator with NIM

To install Adabas System Coordinator to run with the TP system NIM you must do the following:

1. A *multi-TCB* type runtime control must be defined for the NIM job. Multi-TCB type option is on the second screen of selectable runtime control types…select *more choices for type* when adding the runtime control to see the *multi-TCB* choice.

2. Adjust and use the sample job CORI080N to link your re-entrant Adabas link module with the System Coordinator components.

3. Name the supplied NIMUEXC as the NIM user exit C.

4. Name the supplied NIMUEXD as the NIM user exit D.

5. An additional module (CORXNIM) is needed (use sample job CORI090) if any of the following is true…

   ■ You have another NIM exit C and/or D. Name these other exit modules here so that System Coordinator will invoke them after the actual Coordinator equivalents have run.

   Here is an example of how to name these other exit modules in the CORI090 job:

   ```
   //SYSIN    DD *
           CORMNIM EXITC=TESTEXC,    <== ENTER NIM EXIT C MODULE NAME
                   EXITD=TESTEXD     <== ENTER NIM EXIT D MODULE NAME
   /*
   ```

   ■ Your version of NIM pre-dates NIM 4.5.3. For these earlier releases…

   i. Identify one of the four GlobMap user words that is available for use by System Coordinator.
   ii. Identify one of the four TaskMap user words that is available for use by System Coordinator (it does not have to be the same number as the GlobMap).

   Here is an example of how to identify the GlobMap and TaskMap user word fields in the CORI090 job:

   ```
   //SYSIN    DD *
           CORMNIM GBLWORD=GBLUWRD3, <== GLOBAL USER WORD FOR USE BY COR
                   TSKWORD=TSKUSRW2   <== TASK USER WORD FOR USE BY COR
   /*
   ```

6. The System Coordinator load library must be added to the STEPLIB concatenation of your NIM job.

7. The load libraries for the additional products being used (Fastpath, Vista, Transaction Manager etc) must be added to the STEPLIB concatenation for your NIM job.

8. Restart NIM.

Implementing Adabas System Coordinator for zIIP

This documentation describes the implementation of Adabas System Coordinator for zIIP.

- Prerequisites
- Libraries
- License
- Implementation Steps
- Messages and Codes

Prerequisites

zIIP support by Adabas System Coordinator requires that Adabas for zIIP is installed at your site.

Adabas for zIIP relevant prerequisites are described in *Prerequisites* in *Installing Adabas for zIIP* in the *Installation for z/OS* Adabas documentation.

The following table describes the additional Add-on product maintenance for supporting Adabas System Coordinator for zIIP:

Add-on Product	Fix number
Adabas Fastpath	None required.
Adabas SAF Security	AX822013
Adabas Transaction Manager	AT822035
Adabas Vista	None required.

Libraries

Mainframe License Check version 1.3.3 is the minimum MLC version required for the Adabas for zIIP license check performed by the Adabas System Coordinator daemon when running `ZIIP=YES` (see also **License** below).

For up to date information on the required MLC version please refer to the Adabas documentation appropriate for the version of Adabas you are running.

License

An Adabas System Coordinator Daemon that is to run with zIIP activated requires the availability of the Adabas for zIIP license file (AZPAD). If the AZPAD license is not provided or erroneous, the Daemon will run with zIIP deactivated.

The license can be made available to the Adabas System Coordinator Daemon in exactly the same way it is made available to an Adabas nucleus, that is, as a load module with the name AZPADLIC.

Alternatively, the license file can be referred to by a "DDLAZPAD" DD statement in the Daemon job/started task. This is a fall back in case the AZPADLIC module cannot be loaded.

Refer to the licensing information in the Adabas documentation for further information.

Implementation Steps

> **After following the standard** Installation Procedure Steps**, perform the following additional steps to implement Adabas System Coordinator for zIIP**

1   Modify your Daemon job/started task as follows:

   ▪ If you are running Adabas for zIIP version 8.4 or below, replace the standard Adabas library in the Daemon job/started task with the AZPvrs library installed at your site.

   ▪ Include the MLCvrs library for licensing support.

   ▪ Include your license module or dataset (see **License** above)

   For example:

```
//SYSCO1 PROC
//*-------------------------------------------------------------------*
//* System Coordinator SYSCO Vv.r.s. STARTUP *
//*-------------------------------------------------------------------*
//SYSCO1   EXEC PGM=SYSCO,REGION=0M,TIME=1440
//STEPLIB  DD DISP=SHR,DSN=SAG.CORvrs.LOAD
//         DD DISP=SHR,DSN=SAG.ADAvrs.LOAD
//         DD DISP=SHR,DSN=SAG.MLCvrs.LOAD
//         DD DISP=SHR,DSN=your.azpadlic.load.library
//SYSUDUMP DD SYSOUT=*
//CORDUMP  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//DDPRINT  DD SYSOUT=*
//DDCARD   DD DISP=SHR,DSN=SAG.CORvrs.SRCE(CORCNTL)
//*
```

2   Specify ZIIP=YES in the Daemon Runtime Parameters.

3   Start the Adabas System Coordinator Daemon

Messages and Codes

New messages and user abend codes may be observed when running the Adabas System Coordinator Daemon with zIIP activated.

The description of these new messages and codes is organized in the following parts:

ADAI* - ADAIOR System Messages	Describes ADAIOR system messages.
ADAZ* - ADAZIP System Messages	Describes ADAZIP system messages.
CORD* - Adabas System Coordinator Daemon Messages	Describes messages issued by the Adabas System Coordinator daemon.
MLC* - Software GmbH Mainframe Licensing Error Messages	Describes error messages issued by Software GmbH's mainframe licensing software and the license utility.
User Abend Codes	Describes user abend codes.

For ADAI*, ADAZ*, MLC* and User Abend Codes, refer to the corresponding sections of the appropriate Adabas *Messages and Codes* documentation for the version of Adabas you are using.

6 Verifying the Installation

At the end of the installation process, you can use Adabas System Coordinator Online Services (SYSCOR) to check for successful initialization.

Verify Client Component

Client component installation can be verified by performing the following steps:

≫ **To verify client component installation:**

1   Log on to SYSCOR and select option 3 from the main menu.

2   From the Special Services menu, select option 1 to verify that the Adabas System Coordinator is correctly installed.

     A message is displayed confirming successful verification.

     If an error occurs, various messages may be displayed; for more information, see the section Messages and Codes. The following are the most likely causes of an error:

     ▪ The Adabas client (link module) in use does not include the Adabas System Coordinator client component CORS*nn*.

     ▪ The Adabas System Coordinator kernel phase (module) CORKRN is not available to the job.

Verify Adabas System Coordinator Daemon Communication

This step is only required if you intend to use the Adabas System Coordinator daemon to manage clustered applications.

≫ **To verify Adabas System Coordinator communication:**

1   Define the System Coordinator group and member(s) for the daemon(s) you are running. For more information, refer to the section SYSCOR Administration.

2   Define a job parameter for the clustered application, specifying the group name defined in step (1).

3   Start the required Adabas System Coordinator daemon(s).

4   Start, or restart, the clustered TP application.

5   Log on to SYSCOR and select option 2 from the Special Services menu to verify that a clustered TP application can communicate with its Adabas System Coordinator daemon.

A message is displayed confirming successful communication.

Verify the Database Component

⟫ **To verify the database component:**

1   Modify the database startup job control to include the load library containing the Adabas System Coordinator kernel module CORKRN.

2   Modify the database startup parameters to include `FASTPATH=YES,` `VISTA=YES`, or both.

3   Start the database.

The following message is displayed on startup:

POP000I Adapop Vv.r.s initialised, EP=address1 CIB=address2