

Adabas Transaction Manager

Adabas Transaction Manager Installation

Version 8.1.2

June 2008

Adabas Transaction Manager

This document applies to Adabas Transaction Manager Version 8.1.2 and to all subsequent releases.

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

Copyright © Software AG 2008. All rights reserved.

The name Software AG, webMethods and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. Other company and product names mentioned herein may be trademarks of their respective owners.

Table of Contents

1	Adabas Transaction Manager Installation	1
2	Installation Prerequisites	3
	Operating Systems	4
	Software Prerequisites	4
3	Before You Install	9
	Actions Required Prior to Installation	10
	Actions Required Prior to Installation Verification	10
4	Upgrading From Earlier Versions	11
	Upgrading From Version 7.4 To Version 8.1	12
	Upgrading From Version 7.5 To Version 8.1	13
5	Installation Procedure	15
6	Installing Adabas Transaction Manager for z/OS	17
	The Installation Tape	18
	Installation Checklist	18
	Installation Procedure	19
7	Installing CICS Resource Manager Interface for z/OS	27
	Installation Overview	28
	Installation Procedure	28
8	Installing RRMS Interface Under z/OS	31
	Installation Checklist	32
	Installation Procedure	32
9	Installing Adabas Transaction Manager for z/VSE	35
	The Installation Tape	36
	Installation Checklist	36
	Installation Procedure	37
10	Installing CICS Resource Manager Interface for z/VSE	43
	Installation Checklist	44
	Installation Procedure	44
11	Installing Adabas Transaction Manager for BS2000	49
	The Installation Tape	50
	Installation Checklist	50
	Copying the Tape Contents to a BS2000/OSD Disk	51
	Installation Procedure	52
12	Verifying the Installation	57
	Verify Basic Operation of the Adabas Transaction Manager Client Proxy	58
	Verify Communication with the Transaction Manager	58
	Verify Two-Phase Commit Operation	59

1 Adabas Transaction Manager Installation

This document describes how to install Adabas Transaction Manager using installation jobs that are:

- generated by the Software AG System Maintenance Aid (SMA); or
- taken from the job library on the installation tape and manually customized.

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

Each step of the installation procedure is associated with the number of an installation job generated by SMA. If you are not using SMA, a sample job with the same number is provided in the JOBS library on the installation tape. You will need to adapt the sample job to your requirements. Installation job numbers are preceded by the product code ATM; for example, ATMI050.

- [Installation Prerequisites](#)
- [Before You Install](#)
- [Upgrading from Earlier Versions](#)
- [Installation Procedures](#)
- [Verifying the Installation](#)

2 Installation Prerequisites

- Operating Systems 4
- Software Prerequisites 4

Operating Systems

Adabas Transaction Manager Version 8.1.2 is compatible with the following operating systems:

- z/OS
- z/VSE
- BS2000

Software Prerequisites

- [Adabas](#)
- [Adabas Load Library Considerations](#)
- [Adabas Required Data Sets](#)
- [Adabas System Coordinator](#)
- [Entire Net-Work](#)
- [Natural](#)
- [Using Complete, RRMS and CICS Syncpoint Manager](#)

Adabas

Adabas Transaction Manager Version 8.1.2 can be used with databases running under any supported level of Adabas Version 7.4 or above.



Note: If you are using Adabas Version 8.1.1, ensure that the following Adabas fixes are applied: AO811002, AN811021, AN811023, AN811024.

Adabas Load Library Considerations

- [Using Adabas Version 8.1](#)
- [Using Adabas Version 7.4 in z/OS Systems](#)

- Using Adabas Version 7.4 in z/VSE and BS2000 Systems

Using Adabas Version 8.1

Using Adabas Transaction Manager Version 8.1 with Adabas Version 8.1 could not be more simple. Use the ADA81x load library along with the ATM81x and COR81x libraries.

Using Adabas Version 7.4 in z/OS Systems

Using Adabas Transaction Manager Version 8.1 with Adabas Version 7.4 in z/OS systems requires an additional library (currently, Adabas 7.4 LX08) in some cases, to enable various DTP capabilities within the product. The table below clarifies the use of libraries for the various components:

	DTP=TM (Transaction Manager)	DTP=RM (Resource Manager)	DTP=NO (Non-RM Database)
Classic Adabas	ATM81x LX ADA74x	ATM81x ADA74x	ATM81x ADA74x
Adabas Cluster Services	Not permitted	ATM81x LX ADA74x	ATM81x ADA74x
Adabas Parallel Services	Not permitted	ATM81x LX ADA74x	ATM81x ADA74x

In the above table, "LX" indicates that you should concatenate the LX08 library ahead of the ADA74x library in the STEPLIB of the job's JCL.

When a component is converted from Adabas 7.4 to Adabas 8.1 you must stop using the LX library for it.

Before you use the LX library for the first time with a Cluster Services or Parallel Services database, you should run the ADACNV utility which is supplied in the LX library, specifying the following as DDKARTE input:

```
ADACNV   CONVERT   TOVERS=75
```

If you subsequently revert to using the standard Adabas 7.4 library, either because the cluster is now to be run with `DTP=NO`, or because the database is no longer to execute as part of a cluster, you should convert the database back to standard version 7.4 format by running `ADACNV`, as supplied in the LX library, with the following `DDKARTE` input:

```
ADACNV      CONVERT  TOVERS=74
```



Caution: Failure to follow the above instructions can result in various problems relating to the PPT area of the Associator dataset.

Using Adabas Version 7.4 in z/VSE and BS2000 Systems

The Transaction Manager (`DTP=TM`) may not be run under Adabas Version 7.4; it must be run under Adabas Version 8.1.

`DTP=RM` databases may run under Adabas Version 7.4.

Adabas Parallel Services databases running under Version 7.4 must run with `DTP=NO`. If you want an Adabas Parallel Services database to run with `DTP=RM`, it must run under Version 8.1.

Adabas Required Data Sets

This section provides an overview of the required Adabas data sets when using Adabas Transaction Manager.

- [ATM Transaction Manager JCL Requirements](#)
- [Database JCL Requirements](#)

ATM Transaction Manager JCL Requirements

The following table shows the dataset definitions that are essential for correct execution of the ATM transaction manager. Other datasets, such as command log, may be specified. For details of these, refer to the Adabas documentation.

Dataset	Link Name	More Information
Load library		ADA74x library or ADA81x library; COR81x library, ATM81x library Note: Check the <i>Release Notes</i> for details of any special dependencies.
Associator	DDASSORn	
Data Storage	DDDATARn	
Work	DDWORKR1	
ATM log	ATMLOG1/2	Optional, depending on TMLOG parameter value

Dataset	Link Name	More Information
ADARUN parameters	SYSDTA/ DDCARD	DTP=TM, and other operational parameters
ADARUN messages	SYSOUT/ DDPRINT	

Database JCL Requirements

The following table shows the dataset definitions that are essential for correct execution of a single database or cluster that is to execute as a resource manager, under the transactional control of ATM. Other datasets, such as command log, may be specified. For details of these, refer to the Adabas documentation.

Dataset	Link Name	More Information
Load library		ADA74x library or ADA81x library; COR81x library, ATM81x library Note: Check the <i>Release Notes</i> for details of any special dependencies.
Associator	DDASSORn	
Data Storage	DDDATARn	
Work	DDWORKR1	
Work part 4	DDWORKR4	Recoverable information about global transactions. Note: Check the <i>Release Notes</i> for details of any special dependencies.
ADARUN parameters	SYSDTA/ DDCARD	DTP=RM, and other operational parameters
ADARUN messages	SYSOUT/ DDPRINT	

Adabas System Coordinator

Adabas Transaction Manager Version 8.1.2 requires that you first install the Adabas System Coordinator Version 8.1. Refer to the Adabas System Coordinator documentation for more information.

Entire Net-Work

A supported version of Entire Net-Work is required for communication between Adabas Transaction Manager instances in different system images. Refer to the Entire Net-Work documentation for more information.

Natural

Natural is required by the Online Services application SYSATM.

Any supported level of Natural Version 4.1 or above can be used. Refer to the Natural documentation for more information.

Using Com-plete, RRMS and CICS Syncpoint Manager

The Adabas Transaction Manager client proxy currently supports only one Adabas SVC per client session under Com-plete.

For cross-vendor coordination of distributed transactions, Adabas Transaction Manager supports the following IBM asynchronous exit drivers: RRMS and CICS Syncpoint Manager.

3 Before You Install

- Actions Required Prior to Installation 10
- Actions Required Prior to Installation Verification 10

This section describes actions which must be taken prior to performing Adabas Transaction Manager installation.



Note: If you are upgrading from Adabas Transaction Manager Version 7.4/7.5, refer to the section [Upgrading from Earlier Versions of ATM](#) before proceeding with the installation.

Actions Required Prior to Installation

The following actions need to be taken prior to installation:

- reserve a unique DBID for the transaction manager in each operating system image
- identify the TP system library names to be used
- identify the Adabas library names to be used
- install Adabas Version 7.4.3 or above
- install Adabas System Coordinator Version 8.1

Actions Required Prior to Installation Verification

To verify the installation, two databases running with `ADARUN DTP=RM` are required:

- one database containing a standard Employees file, and
- one database containing a standard Vehicles file

During the verification process, views of these files must be available in the Natural environment in which Adabas Transaction Manager Online Services is to be run.

4 Upgrading From Earlier Versions

- Upgrading From Version 7.4 To Version 8.1 12
- Upgrading From Version 7.5 To Version 8.1 13

Upgrading From Version 7.4 To Version 8.1

This section describes actions which must be taken when upgrading from Adabas Transaction Manager Version 7.4.

- [Required Maintenance](#)
- [Recovery Database](#)
- [Upgrading Online Services](#)
- [Upgrading Transaction Managers, Resource Managers and Client Jobs](#)
- [Upgrading Client Runtime Controls](#)

Required Maintenance

If you plan to run components of ATM Version 8.1 alongside components of ATM Version 7.4, you should first apply fix AT742140 to your ATM 7.4 library. This fix allows Version 7.4 transaction managers and Version 8.1 transaction managers to inter-operate, and jointly manage distributed transactions. For further information about inter-operability, please refer to the *Release Notes, Inter-operating with Earlier ATM Versions*.

Recovery Database

Beginning with Version 7.5 of Adabas Transaction Manager, the PRR, STJ and ET data files must reside in the Adabas Transaction Manager's own database, rather than in a separate recovery database. If you used a separate recovery database with ATM 7.4, you can simply unload these files from your recovery database, and load them into the Transaction Manager's database, using the same file numbers.

Upgrading Online Services

The Online Services application (SYSATM) delivered with Adabas Transaction Manager Version 8.1 can be installed alongside the Version 7.4 equivalent. The installation process automatically loads the application into a library with a version-dependent name. The MENU program automatically invokes the most recent installed version, but a function key (PF12) can be used to switch to another installed version. If you plan to run versions 8.1 and 7.4 in parallel, use this feature to ensure that the correct version of SYSATM is used in conjunction with a particular ATM transaction manager or application environment.

Upgrading Transaction Managers, Resource Managers and Client Jobs

Once the Online Services application has been upgraded to Version 8.1, you can begin to upgrade other components. It is not necessary to upgrade all transaction managers at the same time, since the two versions of manager are capable of interacting fully. The following sequence is suggested for each Net-work node:

- Install an ATM Version 8.1 transaction manager alongside the Version 7.4 transaction manager, using a different database ID. The new manager's database files can be created by taking copies of the existing manager's files following a clean shut-down. Run the two managers in parallel until the upgrade process has been completed.
- Upgrade Resource Managers. Replace the ATM 7.4 load library with the Version 8.1 library. It is not necessary to upgrade all resource managers at the same time. Note that the parameter setting `DTP=ET` is not supported beginning with ATM 7.5, so any databases that run with this setting should be changed to run either with `DTP=NO` or with `DTP=RM`. Before converting a cluster database to ATM 8.1, refer to the section Configuration.
- Upgrade client jobs and TP systems. Replace the Version 7.4 ATM load library with the Version 8.1 library.

Upgrading Client Runtime Controls

Beginning with Version 7.5, every client job or TP system that uses Adabas Transaction Manager must be associated with an Adabas System Coordinator group. This association is made by means of a new client runtime control. When you create new runtime controls using Version 8.1 Online Services, you will be forced to provide a value for this control. However, runtime controls that were created using Version 7.4 Online Services must be modified, using Version 8.1 Online Services, so that a System Coordinator Group name can be provided. If a job's ATM runtime controls specify that ATM processing is required, but no System Coordinator Group name is present, a warning message will be issued, the ATM client proxy will fail to initialize, and ATM will take no part in processing the job.

Upgrading From Version 7.5 To Version 8.1

This section describes actions which must be taken when upgrading from Adabas Transaction Manager Version 7.5.

- [Required Maintenance](#)
- [Upgrading Online Services](#)

- [Upgrading COR Daemons, Transaction Managers, Resource Managers and Client Jobs](#)

Required Maintenance

If you plan to run components of ATM Version 8.1 alongside components of ATM Version 7.5, you should first apply fix AT751035 to your ATM 7.5 library. This fix allows Version 7.5 transaction managers and Version 8.1 transaction managers to inter-operate, and jointly manage distributed transactions. For further information, please refer to the *Release Notes*, *Inter-operating with Earlier ATM Versions*.

Upgrading Online Services

The Online Services application (SYSATM) delivered with Adabas Transaction Manager Version 8.1 can be installed alongside the Version 7.5 equivalent. The installation process automatically loads the application into a library with a version-dependent name. The MENU program automatically invokes the most recent installed version, but a function key (PF12) can be used to switch to another installed version. If you plan to run versions 8.1 and 7.5 in parallel, use this feature to ensure that the correct version of SYSATM is used in conjunction with a particular transaction manager or application environment.

Upgrading COR Daemons, Transaction Managers, Resource Managers and Client Jobs

Once the Online Services application has been upgraded to Version 8.1, you can begin to upgrade other components. It is not necessary to upgrade all transaction managers at the same time, since the two versions of manager are capable of interacting fully. The following procedure is suggested for each Net-work node.

- Upgrade the ATM transaction manager from Version 7.5 to Version 8.1. When a transaction manager is upgraded, its local COR daemon must be upgraded at the same time, to use the ATM Version 8.1 load library. At the same time, *DTP=RM* databases that run under the same ID table should be upgraded: replace the ATM 7.5 load library with the Version 8.1 library.
- Upgrade client jobs and TP systems. Replace the Version 7.5 ATM load library with the Version 8.1 library.

5 Installation Procedure

This section describes the procedure for Adabas Transaction Manager installation:

Installation for z/OS

- [Installing Adabas Transaction Manager for z/OS](#)
- [Installing CICS Resource Manager Interface for z/OS](#)
- [Installing RRMS Interface for z/OS](#)

Installation for VSE/ESA

- [Installing Adabas Transaction Manager for z/VSE](#)
- [Installing CICS Resource Manager Interface for z/VSE](#)

Installation for BS2000

- [Installing Adabas Transaction Manager for BS2000](#)

6 Installing Adabas Transaction Manager for z/OS

■ The Installation Tape	18
■ Installation Checklist	18
■ Installation Procedure	19

The Installation Tape

Review the *Report of Tape Creation* that accompanies the release package before restoring the release data to disk.

The installation tape contains the following datasets in the sequence indicated in the report:

Dataset	Contents
ATM vrs .JOBS	Installation jobs, utility jobs
ATM vrs .LOAD	Executable modules
ATM vrs .SYS5	ADALOD input for the ATM PRR file
ATM vrs .SYS6	ADALOD input for the ATM STJ file
ATM vrs .SYS7	ADALOD input for the user proxy log (UPL) file
ATM vrs .SYS8	ADALOD input for the ET data file
ATM vrs .INPL	Online Services application
ATM vrs .ERRN	Online Services messages

where vrs is the version, revision, and system maintenance level of the product. Use the comments in the jobs when modifying them to conform to site requirements.

Installation Checklist

The steps needed for a successful installation are as follows:



Note: The checklist and Installation Procedure indicate how to install a new ATM environment. If you need to upgrade an environment from an earlier version of ATM, refer to [Upgrading from Earlier Versions of ATM](#) for important additional information..

Step	Description	Job Name
	Before proceeding with the installation, ensure that all necessary preparation has been completed as described in section Before You Install .	
1	Restore the libraries from the installation tape.	
2	Create the transaction manager's database.	ATMI030
3	Load the transaction manager files to the transaction manager's database.	ATM1050
4	INPL the Online Services application.	ATM1061
5	Define the default job client runtime controls for use by the transaction manager client proxy.	
6	Prepare the ATM transaction manager job.	ATMI200

Step	Description	Job Name
7	If required, define an Adabas System Coordinator file for migrated transaction records.	
8	Enable databases for two-phase commit.	
9	Prepare the Adabas System Coordinator daemons to support ATM.	
10	Make Adabas Transaction Manager components available as required.	

Installation Procedure

Following is the general Adabas Transaction Manager installation procedure. The actual installation depends on your particular requirements and the specific contents of the release package provided by Software AG 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 Software AG technical support for assistance.

Step1: Copying the Tape 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 Tape to Disk](#)
- [Step 1b: Modify COPY.JOB](#)
- [Step 1c: Submit COPY.JOB](#)



Note: If the datasets for more than one product are delivered on the tape, the dataset COPY.JOB contains the JCL to unload the datasets for all delivered products from the tape 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 Tape to Disk

The data set COPY.JOB (label 2) contains the JCL to unload all other existing data sets from tape 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 tape to your disk.

Step 2: Create the Transaction Manager Database (Job ATMI030) (SMA Job I030, Steps 4190,4191)

The transaction manager database holds recovery information and, depending on the setting of the `TMETDATA` parameter, `ET` data, but no other operational data. The remaining files are for occasional use for special purposes – to store diagnostic log information obtained by the transaction manager client proxy, and to hold details of transactions that were forcibly terminated. The size of this database depends mainly on the number of transactions that might be in progress at the same time, and the total amount of `ET` data, if any, that the ATM manager will need to store.

▶ to create the database:

- Use sample job ATMI030, modified to meet your requirements, to establish the database.

Initially, define a database with the following component sizes:

```
DATA: 250 cylinders  
ASSO: 50 cylinders  
WORK: 50 cylinders
```


TEMP: 20 cylinders
 SORT: 20 cylinders

Enough Data Storage space in the recovery file must be allocated to accommodate the maximum number of concurrently active transactions. The following can be used as a guide for calculating the required Data Storage space:

x times y is the required number of Data Storage blocks,

where x is the maximum number of concurrent transactions/branches at the local transaction manager, and y is the average number of compressed recovery records (PRRs) per Data Storage block. Assume that a compressed PRR requires about 2300 bytes if the transaction has 2000 bytes (compressed) of ET data, and 300 bytes if there is no ET data. It is recommended to allow a safety margin of about 50 percent extra space.

Standard block sizes are adequate for most sites. However, if transactions which perform changes to many databases are to be used, it may be necessary to increase the Data Storage block size to accommodate large recovery records.

No ET data will be stored in this database's checkpoint file. The following parameters should be adequate for the checkpoint file.

```
MAXISN=10,000
DSSIZE=100B
NISIZE=20B
UISIZE=8B
```

Use the normal reporting utilities to determine whether space shortages are likely to occur.

Step 3: Load the Transaction Manager Files into the Transaction Manager's Database (Job ATMI050) (SMA Job I050, Step 4190)

The recovery record file, suspect transaction file, proxy log file and ET data file have numbers 5, 6, 7 and 8, respectively. These file numbers are fixed. Since no application files should be loaded into this database, there should be no conflict.

If ET data is to be stored in the transaction manager's database (TMETDATA=ATM), define file 8 with MAXISN large enough to allow an ISN for each ETID that will ever be used in this Adabas Transaction Manager's local system. A decompressed ET data record occupies about 2 kilobytes. If you choose to store ET data in target databases (TMETDATA=TARGETS), file 8 will not be used. For more information, see TMETDATA parameter.

▶ to load the transaction manager files into the transaction manager's database:

- Use job ATMI050 to define the recovery record, suspect transaction, proxy log and ET data files in the ATM transaction manager's database.

The suggested sizes are adequate for the purposes of testing. You can use the normal reporting utilities to check whether your system needs larger files.

Step 4: Install the Online Services Application SYSATM (Job ATMI061) (SMA Job I061, Steps 4190,4193)

The Adabas Transaction Manager Online Services application is delivered on the installation tape as a NATUNLD dataset created under Natural. The application programs are stored in libraries SYSATM and SYSMT_{vrs} where *vrs* is the version, revision, and system maintenance level of the product.

▶ to install the Online Services application SYSATM:

- 1 Use a Natural utility to install the Online Services application and ERRLODUS to install its messages.
- 2 If Natural Security is installed, define the libraries SYSATM and SYSMT_{vrs} (where *vrs* is the version you are installing, for example 812) and protect as required. You may define MENU as the startup transaction for SYSATM. DO NOT define a startup transaction for SYSMT_{vrs}.

Step 5: Define the Default Client Runtime Controls for the Transaction Manager Client Proxy

The default client runtime controls for the transaction manager client proxy must be defined. For further information, see Online Services, Client Runtime Controls. See also the section Adabas Transaction Manager Components.



Note: If you are upgrading from a previous version of ATM, you should first have installed Version 8.1 of Adabas System Coordinator. As part of this process, you should have defined a new, empty system file, to contain system data such as client runtime controls. When using SYSCOR for the first time with this new file, you would have been given the option of converting data, including ATM's client runtime controls, from a previous version of system file. Refer to the *Adabas System Coordinator* documentation for details. If you did not carry out this conversion, and you wish to use your existing client runtime controls for ATM, log on to SYSATM now, and you will be guided through the process of converting them.



Note: If you are upgrading from ATM Version 7.4, you must modify each set of client runtime controls by providing a value for the Adabas System Coordinator Group Name.

▶ to define the default client runtime controls for the transaction manager client proxy:

- 1 Start a Natural session and log on to the newly installed SYSATM application. Invoke the Menu function. If you have not defined LFILE 152 for your Natural session, a message will prompt you to define it now. Select Systems Settings, then option LFILE 152 Maintenance. Enter the Database ID and file number of the system file that you created when you installed the Adabas System Coordinator. Enter values for the remaining input fields, and confirm your definition. Now return to the main menu and select the Client Runtime Controls option.

For a detailed description of the following step, refer to Online Services, Client Runtime Controls.

- 2 On the Client Runtime Controls screen, press PF10 to add a new default definition. Select an appropriate job type and press Enter. On the next screen, enter *DEFAULT as the job name, and press PF5. Now you can specify the default ATM runtime control settings for the selected job type. Make sure that you set the control Adabas Transaction Manager to ON. Set the SVC number that will be used by the Adabas Transaction Manager that you are installing. You must also set the name of the Adabas System Coordinator group in which your client jobs and TP systems will execute. The provided default values for the remaining client runtime controls should be satisfactory for the purpose of completing and verifying the installation. You can review and change them as required once you have completed and verified the installation.

Step 6: Prepare the ATM Transaction Manager Job (SMA Job I200, Step 4190)

Because the ATM transaction manager executes as a special kind of Adabas nucleus, its JCL is similar to that of a normal Adabas nucleus.

▶ to prepare the ATM transaction manager job:

- 1 Make any necessary site-specific changes to ATMI200 so that it can be used as a procedure for a started task.
- 2 Ensure that any ADALNK module that can be loaded by the ATM job does not have the Adabas System Coordinator stub linked into it.
- 3 Specify ADARUN DTP=TM.
- 4 Specify ADARUN TMLOG=NEVER for the purpose of verifying the installation. Once the verification process has been completed, reconsider this parameter setting.
- 5 Specify ADARUN NONDES=YES. This is the default setting.

Set other ADARUN parameters as you would for a normal database. For initial testing, the default limits and pool and buffer sizes should be sufficient. You can use the sample parameters in the JOBS library, member TMPARMS, as a basis for the transaction manager's ADARUN parameters.

See the section Parameters for a description of the ADARUN parameters that affect Adabas Transaction Manager operation.



Note: An Adabas Transaction Manager must be run in each operating system image that will participate in distributed transaction processing (DTP). If more than one Adabas SVC is used in the same system image, an Adabas Transaction Manager must be run under each Adabas SVC that will participate in DTP.

Step 7: Define an Adabas System Coordinator File

If you plan to use ATM in a multi-system dynamic transaction routing (DTR) environment, you might first need to define an Adabas System Coordinator file (SYSCO file) to hold migrated transaction records (MTRs) for the ATM managers that will execute alongside the Adabas System Coordinator daemons in the group. This file is needed if you plan to use a multi-system DTR environment in which a client who is not at global transaction status can be migrated from one system image to another (for example, a CICSplex cluster with the ATM client runtime control `TransactionModel` set to `DYNAMIC`).

For detailed instructions about creating and defining a SYSCO file, refer to the Adabas System Coordinator documentation. The following paragraphs provide information that you will need when creating the file and making the appropriate definition in SYSCOR.

► to define a SYSCO file to contain migrated transaction records:

- 1 Define a standard Adabas file to contain MTRs. For the Adabas Transaction Managers in a COR group, a single MTR will be stored for each global transaction whose owner is migrated from one system image to another while the transaction is open; the record will be deleted when the transaction is finally committed or backed out. The following ADALOD parameters should be sufficient for 1000 MTRs at the same time, assuming block sizes of about 4 kilobytes for Data Storage and about 2 kilobytes for Associator:

```
DSSIZE=50B
NISIZE=30B
UISIZE=5B
```

During normal use, you can use the standard Adabas reporting utility to check that your file is of adequate size.

- 2 In the online system, SYSCOR, of the Adabas System Coordinator, navigate to the list of defined System Coordinator groups. Locate the group associated with your Adabas Transaction Manager, mark it with 'F', and press `Enter`.

At the next screen, press `PF10` to add a new SYSCO File definition. Enter the MTR file definition in the pop-up window, as follows, specifying a DBID and file number that are suitable for your installation:

```
07:57:00          Add File Definition          2006-04-19
                                          C11270M2

Group Name: CORATMGP

File Name: ATMMTR      DB ID: 135      File Number: 175      SVC: 254
```

```

Description: ATM MIGRATED TRAN RECORDS

Press PF5 to confirm

Command ==>
                                PF1 Help    PF3 Exit    PF5 Add

```

ATM's MTR file must be defined with SYSCO file name ATMMTR.



Note: If you do not create a definition for the MTR file using SYSCOR, ATM will assume that you have determined that no MTR file is needed. Even so, an ATM transaction manager might issue a warning message if it finds no definition. If this happens, and you are sure that you do not need an MTR file, you can prevent the message from appearing simply by creating a dummy MTR file definition for the COR group, using SYSCOR.

Step 8: Enable Databases for Two-Phase Commit Processing

▶ to enable databases for two-phase commit processing

- 1 Ensure that the parameter

```
ADARUN DTP=RM
```

is set in every database that participates in global transactions coordinated by Adabas Transaction Manager.

- 2 Add the Adabas System Coordinator load library and the Adabas Transaction Manager load library to the STEPLIB DD concatenation.
- 3 Ensure that the JCL contains a DD statement for DDWORKR4, and that it identifies a suitable WORK dataset.



Note: The WORK 4 dataset should be allocated and formatted in the normal way, using the same block size as the main WORK dataset. It should be at least as large as the LP parameter of the database or cluster. In the case of a Cluster Services or Parallel Services database, the WORK 4 dataset is shared by all members of the cluster.

Step 9: Prepare the Adabas System Coordinator Daemons

The ATM load library should be made available to all SYSCO daemons which will run under the same Adabas SVC as an ATM transaction manager.

The SYSCO daemons must be started with the parameter setting `PRODUCT=ATM`.

Step 10: Make Adabas Transaction Manager Components Available

The ATM load library should be made available:

- in the STEPLIB of batch jobs, TSO procedures, and JCL procedures for IMS/TM systems that will be used by clients or applications that require Adabas Transaction Manager services;
- in either the DFHRPL or the STEPLIB of CICS systems that will be used by clients or applications that require Adabas Transaction Manager services. Software AG recommends that you use DFHRPL. If you are not using the autoinstall feature, see member DFHDUP in the JOBS library for samples of the required CICS program definitions. Please also see the *Adabas System Coordinator* documentation for details of that component's CICS requirements.
- in the COMPLIB of the JCL procedures for Complete systems that will be used by clients or applications that require Adabas Transaction Manager services.

Ensure that the Adabas System Coordinator stub has been linked into the Adabas link modules that will be executed in each of the above environments.

The ATM load library and the Adabas System Coordinator library should be made available:

- in the STEPLIB of Adabas Transaction Managers;
- in the STEPLIB of multiuser database nucleus jobs that will run with `DTP=RM`.



Note: As in previous releases, there is a requirement to use an unmodified ADALNK in certain areas. With Version 8.1 this restriction has been reduced to unmodified ADALNK being needed only when running standalone Adabas utility jobs.

7 Installing CICS Resource Manager Interface for z/OS

- Installation Overview 28
- Installation Procedure 28

If you intend to use Adabas Transaction Manager through the CICS Resource Manager Interface (RMI), use the following installation procedure in conjunction with the installation procedure for the Adabas task-related user exit (TRUE) described in the *Adabas Installation for z/OS* documentation.



Note: CICS/TS Version 1.2 or above is required.

Installation Overview

The steps needed for a successful installation are as follows:

Step	Description
	Before proceeding with the installation, ensure that all necessary preparation has been completed as described in section Before You Install .
1	Perform standard installation procedure for Adabas Transaction Manager, and verify the installation.
2	Follow the steps for the installation of the Adabas TRUE, with the changes described below.
3	Establish CICS definitions for Adabas Transaction Manager RMI programs.
4	Copy the Adabas Transaction Manager RMI modules into a CICS RPL library.
5	Enable the required CICS facilities.
6	Define client runtime controls for the CICS system.
7	Activate the Adabas Transaction Manager CICS Resource Manager Interface (RMI).

Installation Procedure

The following is the general RMI installation procedure. All steps are required for a successful installation.

The actual installation depends on your particular requirements and the specific contents of the release package provided by Software AG 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 Software AG technical support for assistance.

Step 1: Install Adabas Transaction Manager

It is strongly recommended that you first install and verify the execution of the ATM transaction manager and client proxy without activating the RMI.

Step 2: Install the Adabas TRUE (SMA Job I056, Step 0010)

This process is fully described in the *Adabas Installation for z/OS* documentation. However, the installation process for the RMI differs in a few points from that described. Follow the instructions below as you go through the installation process.

▶ **to activate the RMI:**

- 1 Set the following values in the ADAGSET macro:

RMI=YES	activate the CICS Resource Manager Interface (RMI)
TRUE=YES	run as a task-related user exit (TRUE)

- 2 When linking the Adabas command-level link component, you must link in the Adabas System Coordinator stub module.
- 3 When you assemble and link the Adabas task-related user exit, include the Adabas Transaction Manager RMI interface module in the link step. Sample job ATMI080X can be adapted for this purpose.

Step 3: Establish CICS Definitions for the RMI Programs (SMA Job I005, Steps 2211,2212,2230,2231)

▶ **to establish CICS definitions for the RMI programs:**

- Use DFHCSDUP or the CEDA RDO entry panels to add the definitions delivered in member DFHDUP in the supplied JOBS library.



Note: If storage protection is active in your CICS system, EXECKEY(CICS) should be specified for the PROGRAM definitions.

Step 4: Copy RMI Modules into a CICS RPL library

The following programs, supplied in the ATM load library, must be made available in the DFHRPL DD concatenation of your CICS JCL procedure:

ATMRMIR0
ATMRMIRS
ATMORID

Step 5: Enable the Required CICS Facilities

Adabas Transaction Manager uses the External CICS Interface (EXCI) to trigger the re-synchronization of incomplete transactions during restart. Therefore, the CICS system in which you are installing the RMI must have inter-region communication enabled, and must be enabled for EXCI, the external EXEC CICS interface, which requires a generic connection. Refer to the relevant IBM documentation for details.

If your installation uses a security system, such as RACF, ensure that the User ID under which ATM transaction manager executes has authorization to use EXCI.

Step 6: Define Client Runtime Controls for CICS

Use Adabas Transaction Manager Online Services to define the required client runtime controls for the CICS job.

The required controls are described in section Parameters. Some special considerations are necessary when running with the RMI active. Define a set of client runtime controls for the CICS job. Pay particular attention to the `Client-side TransactionManager`, `TransactionModel` and `TransactionControl`, as well as the `GenerateExternalSyncpoint` client controls.

Step 7: Activate CICS Resource Manager Interface

▶ **to activate the CICS Resource Manager Interface:**

- 1 Ensure that the CICS EXCI load library is available in the STEPLIB concatenation of your Adabas Transaction Manager job or JCL procedure.
- 2 Start CICS.
- 3 Ensure that the Adabas System Coordinator daemon is executing.
- 4 Start the ATM transaction manager.

8

Installing RRMS Interface Under z/OS

- Installation Checklist 32
- Installation Procedure 32

If you intend to use Adabas Transaction Manager in conjunction with the Recoverable Resource Management Services (RRMS), use the following installation procedure.

Installation Checklist

The steps needed for a successful installation are as follows:

Step	Description
1	Perform standard installation procedure for Adabas Transaction Manager, and verify the installation.
2	Copy the Adabas Transaction Manager RRMS exits into an LPA library.
3	APF-authorize the ATM transaction manager job.
4	Link the RRMS interface modules to the Adabas Transaction Manager kernel (job ATMI070).
5	Define client runtime controls for the jobs, Complete systems, and IMS systems whose transactions will be controlled by RRS.
6	Verify the system logger and RRS operation.
7	Activate the Adabas Transaction Manager RRMS interface.

Installation Procedure

The following is the general procedure for installing the Adabas Transaction Manager RRMS interface. All steps are required for a successful installation.

The actual installation depends on your particular requirements and the specific contents of the release package provided by Software AG 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 Software AG technical support for assistance.

Step 1: Install Adabas Transaction Manager

It is strongly recommended that you first install and verify the execution of the ATM transaction manager and client proxy without activating the RRMS interface.

Step 2: Copy RRMS Exits into an LPA Library

The Adabas Transaction Manager RRMS exits are contained in the single module ATMRRSEX, which is supplied in the Adabas Transaction Manager load library.

This module must be made available in the LPA of your z/OS system. You can use various third-party tools to place ATMRRSEX in the LPA temporarily. For permanent installation, you should copy the module into a library that appears in the system LPALST concatenation, then IPL the system specifying the CLPA option.

If ATMRRSEX is not available in the LPA when Adabas Transaction Manager attempts to initialize its RRMS interface, system ABEND 052 occurs.

Step 3: APF-Authorize the ATM Transaction Manager Job

RRMS requires that resource managers execute as APF-authorized programs. The ATM transaction manager must run authorized if its RRMS interface is to be used. However, databases that run with `DTP=RM` do not need to be authorized.

Ensure that every library in the STEPLIB concatenation of the ATM transaction manager's JCL procedure is APF-authorized.

Step 4: Link the RRMS Interface Modules to the ATM Kernel (Job ATMI070) (SMA Job I070, Step 4190)

In order for an application's transactions to be coordinated by RRMS, some additional interface modules from IBM must be linked into the Adabas Transaction Manager kernel.

Job ATMI070 can be adapted for this purpose. Unresolved external references can be ignored, provided that the named RRMS modules are included successfully.

Step 5: Create Client Runtime Controls

Use Adabas Transaction Manager Online Services to define the required client runtime controls.

The required controls are described in section Parameters. Some special considerations are necessary when running with the RRMS interface active. Define a set of client runtime controls for those batch jobs, Com-plete systems, and IMS/TM systems that require the transactional control of RRS, and pay particular attention to the `TransactionControl` client control.

Step 6: Verify the System Logger and RRS Operation

Ensure that the system logger is active in your z/OS system and that RRS has been started. Refer to the *IBM z/OS* documentation for details.

Step 7: Activate the Adabas Transaction Manager RRMS Interface

▶ **to activate the RRMS Interface:**

- 1 Specify `TMSYNCMGR=RRMS` in the ATM transaction manager ADARUN parameters.
- 2 Ensure that the Adabas System Coordinator daemon is running.
- 3 Start the ATM transaction manager.

9 Installing Adabas Transaction Manager for z/VSE

■ The Installation Tape	36
■ Installation Checklist	36
■ Installation Procedure	37

The Installation Tape

Review the *Report of Tape Creation* that accompanies the release package before restoring the release data to disk.

The installation tape contains the following datasets in the sequence indicated in the report:

Dataset	Contents
ATMvrs.LIBR	Executable modules and sample jobs
ATMvrs.SYS5	ADALOD input for the ATM PRR file
ATMvrs.SYS6	ADALOD input for the ATM STJ file
ATMvrs.SYS7	ADALOD input for the user proxy log (UPL) file
ATMvrs.SYS8	ADALOD input for the ET data file
ATMvrs.INPL	Online Services application
ATMvrs.ERRN	Online Services messages

where *vrs* is the version, revision, and system maintenance level of the product. Use the comments in the jobs when modifying them to conform to site requirements.

Installation Checklist

The steps needed for a successful installation are as follows:

Step	Description	Job Name
	Before proceeding with the installation, ensure that all necessary preparation has been completed as described in section Before You Install .	
1	Restore the libraries from the installation tape.	
2	Create the transaction manager's database.	ATMI030
3	Load the transaction manager files to the ATM transaction manager's database.	ATM1050
4	INPL the Online Services application SYSATM.	ATM1061
5	Define the default client runtime controls for use by the transaction manager client proxy.	
6	Prepare the ATM transaction manager job.	ATMI200
7	Enable databases for two-phase commit.	
8	Prepare the Adabas System Coordinator daemons to support ATM.	
9	Make Adabas Transaction Manager components available as required.	
10	Define Adabas Transaction Manager to MSHP CORRECT facility.	

Installation Procedure

Following is the general Adabas Transaction Manager installation procedure. The actual installation depends on your particular requirements and the specific contents of the release package provided by Software AG 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 Software AG technical support for assistance.

Step 1: Restore Libraries from the Installation Tape

▶ to restore libraries from the installation tape:

- Use LIBR to unload the library:

Variable	Is the...
<i>cuu</i>	tape unit number
<i>vrs</i>	version, revision, and system maintenance level
<i>tttttt</i>	volume serial number of the installation tape
<i>xx</i>	file spacing information; see the Report of Tape Creation

```
* $$ JOB JNM=LIBREST,CLASS=O,DISP=D
* $$ LST CLASS=A,DISP=H
// JOB LIBREST
// ASSGN SYS006,cuu,VOL=tttttt
// ASSGN SYS005,IGN
// MTC REW,SYS006
// MTC FSF,SYS006,xx
// EXEC LIBR
RESTORE S=SAGLIB.AVivrS:SAGLIB.AVivrS -
TAPE=SYS006 TL=tttttt LIST=Y
/*
// MTC REW,SYS006
/&
* $$ EOJ
```

Step 2: Create the Transaction Manager Database (Job ATMI030)

The transaction manager database holds recovery information and, depending on the setting of the `TMETDATA` parameter, `ET` data, but no other operational data. The remaining files are for occasional use for special purposes – to store diagnostic log information obtained by the transaction manager client proxy, and to hold details of transactions that were forcibly terminated. The size of this database depends mainly on the number of transactions that might be in progress at the same time, and the total amount of `ET` data, if any, that the ATM manager will need to store.

▶ **to create the database:**

- Use sample job ATMI030, modified to meet your requirements, to establish the database.

Initially, define a database with the following component sizes:

DATA: 250 cylinders
ASSO: 50 cylinders
WORK: 50 cylinders
TEMP: 20 cylinders
SORT: 20 cylinders

Enough Data Storage space in the recovery file must be allocated to accommodate the maximum number of concurrently active transactions. The following can be used as a guide for calculating the required Data Storage space:

x times y is the required number of Data Storage blocks,

where x is the maximum number of concurrent transactions/branches at the local transaction manager, and y is the average number of compressed recovery records (PRRs) per Data Storage block. Assume that a compressed PRR requires about 2300 bytes if the transaction has 2000 bytes (compressed) of `ET` data, and 300 bytes if there is no `ET` data. It is recommended to allow a safety margin of about 50 percent extra space.

Standard block sizes are adequate for most sites. However, if transactions which perform changes to many databases are to be used, it may be necessary to increase the Data Storage block size to accommodate large recovery records.

No `ET` data will be stored in this database's checkpoint file. The following parameters should be adequate for the checkpoint file.

```
MAXISN=10,000  
DSSIZE=100B  
NISIZE=20B  
UISIZE=8B
```

Use the normal reporting utilities to determine whether space shortages are likely to occur.

Step 3: Load the Transaction Manager Files into the Transaction Manager's Database (Job ATMI050)

The recovery record file, suspect transaction file, proxy log file and ET data file have numbers 5, 6, 7 and 8, respectively. These file numbers are fixed. Since no application files should be loaded into this database, there should be no conflict.

If ET data is to be stored in the transaction manager's database (`TMETDATA=ATM`), define file 8 with `MAXISN` large enough to allow an ISN for each ETID that will ever be used in this Adabas Transaction Manager's local system. A decompressed ET data record occupies about 2 kilobytes. If you choose to store ET data in target databases (`TMETDATA=TARGETS`), file 8 will not be used. For more information, see `TMETDATA` parameter.

▶ **to load the transaction manager files into the transaction manager's database:**

- Use job ATMI050 to define the recovery record, suspect transaction, proxy log and ET data files in the ATM transaction manager's database.

The suggested sizes are adequate for the purposes of testing. You can use the normal reporting utilities to check whether your system needs larger files.

Step 4: Install the Online Services Application SYSATM (Job ATMI061)

The Adabas Transaction Manager Online Services application is delivered on the installation tape as a NATUNLD dataset created under Natural. The application programs are stored in libraries `SYSATM` and `SYSMTvrs` where *vrs* is the version, revision, and system maintenance level of the product.

▶ **to install the Online Services application SYSATM:**

- 1 Use a Natural utility to install the Online Services application and `ERRLODUS` to install its messages.
- 2 If Natural Security is installed, define the libraries `SYSATM` and `SYSMTvrs` (where *vrs* is the version you are installing, for example 812) and protect as required. You may define `MENU` as the startup transaction for `SYSATM`. DO NOT define a startup transaction for `SYSMTvrs`.

Step 5: Define the Default Client Runtime Controls for the Transaction Manager Client Proxy

The default client runtime controls for the Transaction Manager client proxy must be defined. For further information, see Online Services, Client Runtime Controls. See also the section Adabas Transaction Manager Components.

▶ to define the default client runtime controls for the transaction manager client proxy:

- 1 Start a Natural session and log on to the newly installed SYSATM application. Invoke the Menu function. If you have not defined LFILE 152 for your Natural session, a message will prompt you to define it now. Select Systems Settings, then option LFILE 152 Maintenance. Enter the Database ID and file number of the system file that you created when you installed the Adabas System Coordinator. Enter values for the remaining input fields, and confirm your definition. Now return to the main menu and select the Client Runtime Controls option. For a detailed description of the following step, refer to Online Services, Client Runtime Controls
- 2 On the Client Runtime Controls screen, press PF10 to add a new default definition. Select an appropriate job type and press Enter. On the next screen, enter *DEFAULT as the job name, and press PF5. Now you can specify the default ATM runtime control settings for the selected job type. Make sure that you set Adabas Transaction Manager to ON. Set the SVC number that will be used by the Adabas Transaction Manager that you are installing. You must also set the name of the Adabas System Coordinator group in which your client jobs and TP systems will execute. The provided default values for the remaining controls should be satisfactory for the purpose of completing and verifying the installation. You can review and change them as required once you have completed and verified the installation.

Step 6: Prepare the ATM Transaction Manager Job

Because the ATM transaction manager executes as a special kind of Adabas nucleus, its JCL is similar to that of a normal Adabas nucleus.

▶ to prepare the ATM transaction manager job:

- 1 Make any necessary site-specific changes to ATMI200 so that it can be used as a procedure for a started task.
- 2 Ensure that any ADALNK module that can be loaded by the ATM job does not have the Adabas System Coordinator stub linked into it.
- 3 Specify `ADARUN DTP=TM.`
- 4 Specify `ADARUN TMLLOG=NEVER` for the purpose of verifying the installation. Once the verification process has been completed, reconsider this parameter setting.
- 5 Specify `ADARUN NONDES=YES.` This is the default setting.

Set other ADARUN parameters as you would for a normal database. For initial testing, the default limits and pool and buffer sizes should be sufficient.

See the section Parameters for a description of the ADARUN parameters that affect Adabas Transaction Manager operation.



Note: An Adabas Transaction Manager job must be run in each operating system image that will participate in distributed transaction processing (DTP). If more than one Adabas SVC is used in the same system image, an Adabas Transaction Manager must be run under each Adabas SVC that will participate in DTP.

Step 7: Enable Databases for Two-Phase Commit Processing

▶ to enable databases for two-phase commit processing

- 1 Ensure that the parameter

```
ADARUN DTP=RM
```

is set in every database that participates in global transactions coordinated by Adabas Transaction Manager.

- 2 Add the ATM load library to the LIBDEF SEARCH chain.
- 3 Ensure that the JCL contains a specification for DDWORKR4, and that it identifies a suitable WORK dataset.



Note: The WORK 4 dataset should be allocated and formatted in the normal way, using the same block size as the main WORK dataset. It should be at least as large as the LP parameter of the database or cluster. In the case of a Cluster Services or Parallel Services database, the WORK 4 dataset is shared by all members of the cluster.

Step 8: Prepare the Adabas System Coordinator Daemons

The ATM load library should be made available to all SYSCO daemons which will run under the same Adabas SVC as an ATM transaction manager.

The SYSCO daemons must be started with the parameter setting `PRODUCT=ATM`.

Step 9: Make Adabas Transaction Manager Components Available

The ATM load library should be made available:

- in the LIBDEF SEARCH chain of batch jobs and JCL procedures for IMS/TM systems that will be used by users or applications that require Adabas Transaction Manager services;
- in either the DFHRPL or the LIBDEF SEARCH chain of CICS systems that will be used by clients or applications that require Adabas Transaction Manager services. Software AG recommends that you use DFHRPL. If you are not using the autoinstall feature, see member DFHDUP in the JOBS library for samples of the required CICS program definitions. Please also see the *Adabas System Coordinator* documentation for details of that component's CICS requirements.
- in the COMPLIB of the JCL procedures for Com-plete systems that will be used by users or applications that require Adabas Transaction Manager services.

Ensure that the Adabas System Coordinator stub has been linked into the Adabas link modules that will be executed in each of the above environments.

The ATM load library and the Adabas System Coordinator library should be made available:

- in the LIBDEF SEARCH chain of Adabas Transaction Managers;
- in the LIBDEF SEARCH chain of multiuser database nucleus jobs that will run with `DTP=RM`.



Note: As in previous releases, there is a requirement to use an unmodified ADALNK in certain areas. With Version 8.1 this restriction has been reduced to unmodified ADALNK being needed only when running standalone Adabas utility jobs.

Step 10: Define Adabas Transaction Manager to MSHP CORRECT Facility

Corrections for Adabas Transaction Manager will be distributed as zaps that are to be applied using the MSHP CORRECT facility. Before applying corrections you must define Adabas Transaction Manager to MSHP with the `MSHP ARCHIVE` command (see the following sample job):

```
// JOB ARCHIVE
// OPTION LOG
// EXEC MSHP
ARCHIVE ATMvrs
COMPRISES 9001-ATM-00
RESOLVES 'SOFTWARE AG - ADABAS TRANSACTION MANAGER Vv.r'
ARCHIVE 9001-ATM-00-vrs
RESIDENCE PRODUCT=ATMvrs -
PRODUCTION=saglib.ATMvrs -
GENERATION=saglib.ATMvrs
/*
/&
```

10

Installing CICS Resource Manager Interface for z/VSE

- Installation Checklist 44
- Installation Procedure 44

If you intend to use Adabas Transaction Manager through the CICS Resource Manager Interface (RMI), use the following installation procedure in conjunction with the installation procedure for the Adabas task-related user exit (TRUE) described in the *Adabas Installation for z/OS* documentation.



Note: CICS/ESA Version 4.1 or above or CICS/TS Version 1.2 or above is required.

Installation Checklist

The steps needed for a successful installation are as follows:

Step	Description
	Before proceeding with the installation, ensure that all necessary preparation has been completed as described in section Before You Install .
1	Perform standard installation procedure for Adabas Transaction Manager, and verify the installation.
2	Follow the steps for the installation of the Adabas TRUE, with the changes described below.
3	Establish CICS definitions for Adabas Transaction Manager RMI programs.
4	Copy the Adabas Transaction Manager RMI modules into a CICS RPL library.
5	Enable the required CICS facilities.
6	Define client runtime controls for the CICS system.
7	Activate the Adabas Transaction Manager CICS Resource Manager Interface (RMI).

Installation Procedure

The following is the general RMI installation procedure. All steps are required for a successful installation.

The actual installation depends on your particular requirements and the specific contents of the release package provided by Software AG 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 Software AG technical support for assistance.

Step 1: Install Adabas Transaction Manager

It is strongly recommended that you first install and verify the execution of the ATM transaction manager and client proxy without activating the RMI.

Step 2: Install the Adabas TRUE

This process is fully described in the *Adabas Installation for VSE/ESA* documentation. However, the installation process for the RMI differs in a few points from that described. Follow the instructions below as you go through the installation process.

▶ to activate the RMI:

- 1 Set the following values in the ADAGSET macro:

RMI=YES	activate the CICS Resource Manager Interface (RMI)
TRUE=YES	run as a task-related user exit (TRUE)

- 2 When linking the Adabas command-level link component, you must link in the Adabas System Coordinator stub module.
- 3 When you assemble and link the Adabas task-related user exit, include the Adabas Transaction Manager RMI interface module in the link step. Sample job ATMI080X can be adapted for this purpose.

Step 3: Establish CICS Definitions for the RMI Programs

▶ to establish CICS definitions for the RMI programs:

- Use DFHCSDUP or the CEDA RDO entry panels to add the following definitions to your CSD file:

```
DEFINE PROGRAM(ATMRMIR0) GROUP(ADABAS)
DESCRIPTION(ATM Vvrs RESYNC DRIVER INTERFACE)
LANGUAGE(ASSEMBLER) RELOAD(NO) RESIDENT(YES) USAGE(NORMAL)
USELPACOPY(NO) STATUS(ENABLED) CEDF(NO) DATALOCATION(ANY)
EXECKEY(USER) EXECUTIONSET(FULLAPI)

DEFINE PROGRAM(ATMRMIRS) GROUP(ADABAS)
DESCRIPTION(ATM Vvrs RESYNC DRIVER)
LANGUAGE(ASSEMBLER) RELOAD(NO) RESIDENT(YES) USAGE(NORMAL)
USELPACOPY(NO) STATUS(ENABLED) CEDF(YES) DATALOCATION(ANY)
EXECKEY(USER) EXECUTIONSET(FULLAPI)

DEFINE PROGRAM(ATMORID) GROUP(ADABAS)
```

```
DESCRIPTION(ATM Vvrs PARAMETER OVERRIDER)
LANGUAGE(ASSEMBLER) RELOAD(NO) RESIDENT(YES) USAGE(NORMAL)
USELPACOPY(NO) STATUS(ENABLED) CEDF(YES) DATALOCATION(ANY)
EXECKEY(USER) EXECUTIONSET(FULLAPI)

DEFINE TRANSACTION(TMRS) GROUP(ADABAS)
DESCRIPTION(ATM Vvrs RESYNC DRIVER)
PROGRAM(ATMRMIRS) TWASIZE(128) PROFILE(DFHICST) STATUS(ENABLED)
TASKDATALOC(ANY) TASKDATAKEY(USER) STORAGECLEAR(NO)
RUNAWAY(SYSTEM) SHUTDOWN(DISABLED) ISOLATE(YES) DYNAMIC(NO)
PRIORITY(1) TRANCLASS(DFHTCL00) DTIMOUT(NO) INDOUBT(BACKOUT)
RESTART(NO) SPURGE(NO) TPURGE(NO) DUMP(YES) TRACE(YES)
RESSEC(NO) CMDSEC(NO)
```

—where *vrs* is the version, revision, and SM level of Adabas Transaction Manager.



Note: If storage protection is active in your CICS system, EXECKEY(CICS) should be specified for the PROGRAM definitions.

Step 4: Copy RMI Modules into a CICS RPL library

The following programs, supplied in the ATM load library, must be made available in the LIBDEF concatenation of your CICS JCL procedure:

```
ATMRMIR0
ATMRMIRS
ATMORID
```

Step 5: Enable the Required CICS Facilities

Adabas Transaction Manager uses the External CICS Interface (EXCI) to trigger the re-synchronization of incomplete transactions during restart. Therefore, the CICS system in which you are installing the RMI must have inter-region communication enabled, and must be enabled for EXCI, the external EXEC CICS interface, which requires a generic connection. Refer to the relevant IBM documentation for details.

Step 6: Define Client Runtime Controls for CICS

Use Adabas Transaction Manager Online Services to define the required client runtime controls for the CICS job.

The required controls are described in section Parameters. Some special considerations are necessary when running with the RMI active. Define a set of client runtime controls for the CICS job. Pay particular attention to the `Client-sideTransactionManager`, `TransactionModel` and `TransactionControl`, as well as the `GenerateExternalSyncpoint` client controls.

Step 7: Activate CICS Resource Manager Interface

▶ to activate the CICS Resource Manager Interface:

- 1 Ensure that the CICS EXCI load library is available in the LIBDEF SEARCH concatenation of your Adabas Transaction Manager job or JCL procedure.
- 2 Start CICS.
- 3 Ensure that the Adabas System Coordinator daemon is running.
- 4 Start the Adabas Transaction Manager.

11 Installing Adabas Transaction Manager for BS2000

■ The Installation Tape	50
■ Installation Checklist	50
■ Copying the Tape Contents to a BS2000/OSD Disk	51
■ Installation Procedure	52

The Installation Tape

Review the *Report of Tape Creation* that accompanies the release package before restoring the release data to disk. Information in this report supersedes the information in this documentation.

The installation tape contains the following datasets in the sequence indicated in the report:

Dataset	Contents
ATM vrs .JOBS	ATM installation jobs, utility jobs
ATM vrs .MOD	ATM load modules
ATM vrs .SYS5	ADALOD input for the ATM PRR file
ATM vrs .SYS6	ADALOD input for the ATM STJ file
ATM vrs .SYS7	ADALOD input for the user proxy log (UPL) file
ATM vrs .SYS8	ADALOD input for the ET data file
ATM vrs .INPL	SYSATM objects
ATM vrs .ERRN	SYSATM error messages

where vrs is the version, revision, and system maintenance level of the product. Use the comments in the jobs when modifying them to conform to site requirements.

Installation Checklist

The Adabas System Coordinator must be installed prior to the installation of Adabas Transaction Manager. See the Adabas System Coordinator documentation for detailed information.

Once the required System Coordinator components have been installed, you can proceed to install Adabas Transaction Manager.

After copying the tape contents to disk, the following checklist identifies the steps necessary to complete the installation:

Step	Description	Job Name
	Before proceeding with the installation, ensure that all necessary preparation has been completed as described in section Before You Install .	
1	Create the transaction manager's database.	ATMI030
2	Load the transaction manager files to the ATM transaction manager's database.	ATMI050
3	INPL the Online Services application.	ATMI061
4	Define the default client runtime controls for use by the transaction manager client proxy.	

Step	Description	Job Name
5	Prepare the ATM transaction manager job.	ATMI200
6	Enable databases for two-phase commit.	
7	Prepare the Adabas System Coordinator daemons to support ATM.	
8	Make Adabas Transaction Manager components available as required.	

Copying the Tape Contents to a BS2000/OSD Disk

▶ to copy the tape contents to a BS2000/OSD disk:

- 1 Copy the library SRVnnn.LIB from tape to disk.

This action is not necessary if you have already copied the library SRVnnn.LIB from another Software AG tape. For more information, refer to the element #READ-ME in this library.

The library SRVnnn.LIB is stored on the tape as the sequential file SRVnnn.LIBS containing LMS commands. The current version nnn can be obtained from the Report of Tape Creation. To convert this sequential file into an LMS-library, execute the following commands:

```
/IMPORT-FILE SUPPORT=*TAPE(FILE-NAME=SRVnnn.LIBS, -
/ VOLUME=<volser>, DEV-TYPE=<tape-device>)
/ADD-FILE-LINK LINK-NAME=EDTSAM, FILE-NAME=SRVnnn.LIBS, -
/ SUPPORT=*TAPE(FILE-SEQ=9), ACC-METH=*BY-CAT, -
/ BUF-LEN=*BY-CAT, REC-FORM=*BY-CAT, REC-SIZE=*BY-CAT
/START-EDT
@READ '/'
@SYSTEM 'REMOVE-FILE-LINK EDTSAM'
@SYSTEM 'EXPORT-FILE FILE-NAME=SRVnnn.LIBS'
@WRITE 'SRVnnn.LIBS'
@HALT
/ASS-SYSDTA SRVnnn.LIBS
/MOD-JOB-SW ON=1
/START-PROG $LMS
/MOD-JOB-SW OFF=1
/ASS-SYSDTA *PRIMARY
<tape-device> = device-type of the tape, e.g. TAPE-C4
<volser> = VOLSER of tape (see Report of Tape Creation)
```

- 2 Copy the procedure COPY.PROC from tape to disk

Call the procedure P.COPYTAPE in the library SRV nnn .LIB to copy the procedure COPY.PROC to disk:

```
/CALL-PROCEDURE (SRV $nnn$ .LIB,P.COPYTAPE), -  
/ (VSNT=<volser>, DEVT=<tape-device>)
```

If you use a TAPE-C4 device, you can omit the parameter DEVT.

3 Copy all product files from tape to disk

Enter the procedure COPY.PROC to copy all Software AG product files from tape to disk:

```
/ENTER-PROCEDURE COPY.PROC, DEVT=<tape-device>
```

If you use a TAPE-C4 device, you can omit the parameter DEVT. The result of this procedure is written to the file L.REPORT.SRV.

Installation Procedure

Following is the general Adabas Transaction Manager installation procedure. The actual installation depends on your particular requirements and the specific contents of the release package provided by Software AG 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 Software AG technical support for assistance.

Step 1: Create the Transaction Manager Database (Job ATMI030)

The transaction manager database holds recovery information and, depending on the setting of the TMETDATA parameter, ET data, but no other operational data. The remaining files are for occasional use for special purposes – to store diagnostic log information obtained by the transaction manager client proxy, and to hold details of transactions that were forcibly terminated. The size of this database depends mainly on the number of transactions that might be in progress at the same time, and the total amount of ET data, if any, that the ATM manager will need to store.

▶ to create the database:

- Use sample job ATMI030, modified to meet your requirements, to establish the database.

Initially, define a database with the following component sizes:

DATA: 250 cylinders

ASSO: 50 cylinders

WORK: 50 cylinders

TEMP: 20 cylinders
 SORT: 20 cylinders

Enough Data Storage space in the recovery file must be allocated to accommodate the maximum number of concurrently active transactions. The following can be used as a guide for calculating the required Data Storage space:

x times y is the required number of Data Storage blocks,

where x is the maximum number of concurrent transactions/branches at the local ATM transaction manager, and y is the average number of compressed recovery records (PRRs) per Data Storage block. Assume that a compressed PRR requires about 2300 bytes if the transaction has 2000 bytes (compressed) of ET data, and 300 bytes if there is no ET data. It is recommended to allow a safety margin of about 50 percent extra space.

Standard block sizes are adequate for most sites. However, if transactions which perform changes to many databases are to be used, it may be necessary to increase the Data Storage block size to accommodate large recovery records.

No ET data will be stored in this database's checkpoint file. The following parameters should be adequate for the checkpoint file.

```
MAXISN=10,000
DSSIZE=100B
NISIZE=20B
UISIZE=8B
```

Use the normal reporting utilities to determine whether space shortages are likely to occur.

Step 2: Load the Transaction Manager Files into the Transaction Manager's Database (Job ATMI050)

The recovery record file, suspect transaction file, proxy log file and ET data file have numbers 5, 6, 7 and 8, respectively. These file numbers are fixed. Since no application files should be loaded into this database, there should be no conflict.

If ET data is to be stored in the transaction manager's database (`TMETDATA=ATM`), define file 8 with `MAXISN` large enough to allow an ISN for each ETID that will ever be used in this Adabas Transaction Manager's local system. A decompressed ET data record occupies about 2 kilobytes. If you choose to store ET data in target databases (`TMETDATA=TARGETS`), file 8 will not be used. For more information, see `TMETDATA` parameter.

▶ to load the transaction manager files into the transaction manager's database:

- Use job ATMI050 to define the recovery record, suspect transaction, proxy log and ET data files in the transaction manager's database.

The suggested sizes are adequate for the purposes of testing. You can use the normal reporting utilities to check whether your system needs larger files.

Step 3: Install the Online Services Application SYSATM (Job ATMI061)

The Adabas Transaction Manager Online Services application is delivered on the installation tape as a NATUNLD dataset created under Natural. The application programs are stored in libraries SYSATM and SYSMT vr s where vr s is the version, revision, and system maintenance level of the product.

▶ to install the Online Services application SYSATM:

- 1 Use a Natural utility to install the Online Services application and ERRLODUS to install its messages.
- 2 If Natural Security is installed, define the libraries SYSATM and SYSMT vr s (where vr s is the version you are installing, for example 812) and protect as required. You may define MENU as the startup transaction for SYSATM. DO NOT define a startup transaction for SYSMT vr s.

Step 4: Define the Default Client Runtime Controls for the Transaction Manager Client Proxy

The default client runtime controls for the Transaction Manager client proxy must be defined. For further information, see Online Services, Client Runtime Controls. See also the section Adabas Transaction Manager Components.



Note: If you are upgrading from a previous version of ATM, you should first have installed Version 8.1 of Adabas System Coordinator. As part of this process, you should have defined a new, empty system file, to contain system data such as client runtime controls. When using SYSCOR for the first time with this new file, you would have been given the option of converting data, including ATM's client runtime controls, from a previous version of system file. Refer to the *Adabas System Coordinator* documentation for details. If you did not carry out this conversion, and you wish to use your existing client runtime controls for ATM, log on to SYSATM now, and you will be guided through the process of converting them.

▶ to define the default client runtime controls for the transaction manager client proxy:

- 1 Start a Natural session and log on to the newly installed SYSATM application. Invoke the Menu function. If you have not defined LFILE 152 for your Natural session, a message will prompt you to define it now. Select the function Systems Settings, then option LFILE 152 Maintenance. Enter the Database ID and file number of the system file that you created when you installed the Adabas System Coordinator. Enter values for the remaining input fields, and confirm your definition. Now return to the main menu and select the Client Runtime Controls option. For a detailed description of the following step, refer to Online Services, Client Runtime Controls.

- 2 On the Client Runtime Controls screen, press PF10 to add a new default definition. Select an appropriate job type and press Enter. On the next screen, enter *DEFAULT as the job name, and press PF5. Now you can specify the default ATM runtime control settings for the selected job type. Make sure that you set Adabas Transaction Manager to ON. You must also set the name of the Adabas System Coordinator group in which your client jobs and TP systems will execute. The provided default values for the remaining controls should be satisfactory for the purpose of completing and verifying the installation. You can review and change them as required once you have completed and verified the installation.

Step 5: Prepare the ATM Transaction Manager Job

Because the ATM transaction manager executes as a special kind of Adabas nucleus, its JCL is similar to that of a normal Adabas nucleus.

▶ to prepare the ATM transaction manager job:

- 1 Make any necessary site-specific changes to ATM1200 so that it can be used as a procedure for a started task.
- 2 Ensure that any ADALNK module that can be loaded by the ATM job does not have the Adabas System Coordinator stub linked into it.
- 3 Specify `ADARUN DTP=TM`.
- 4 Specify `ADARUN TMLOG=NEVER` for the purpose of verifying the installation. Once the verification process has been completed, reconsider this parameter setting.
- 5 Specify `ADARUN NONDES=YES`. This is the default setting.

Set other ADARUN parameters as you would for a normal database. For initial testing, the default limits and pool and buffer sizes should be sufficient.

See the section Parameters for a description of the ADARUN parameters that affect Adabas Transaction Manager operation.



Note: An ATM transaction manager job must be run in each operating system image that will participate in distributed transaction processing (DTP). If more than one Adabas ID table is used in the same system image, an Adabas System Coordinator daemon and an ATM transaction manager must be run under each Adabas ID table that will participate in DTP. For further information, see ATM transaction managers and Adabas System Coordinator Groups.

Step 6: Enable Databases for Two-Phase Commit Processing

▶ to enable databases for two-phase commit processing

- 1 Ensure that the parameter

```
ADARUN DTP=RM
```

is set in every database that participates in global transactions coordinated by Adabas Transaction Manager.

- 2 Add ADD-FILE-LINK statements for the Adabas Transaction Manager product library and the Adabas System Coordinator product library to each database job that will run with `DTP=RM`.
- 3 Ensure that the job contains a specification for `DDWORKR4`, and that it identifies a suitable `WORK` dataset.



Note: The `WORK 4` dataset should be allocated and formatted in the normal way, using the same block size as the main `WORK` dataset. It should be at least as large as the `LP` parameter of the database or cluster. In the case of a Cluster Services or Parallel Services database, the `WORK 4` dataset is shared by all members of the cluster.

Step 7: Prepare the Adabas System Coordinator Daemons

The ATM load library should be made available to all SYSCO daemons which will run under the same Adabas ID table as an ATM transaction manager.

The SYSCO daemons must be started with the parameter setting `PRODUCT=ATM`.

Step 8: Make Adabas Transaction Manager Components Available

Add an ADD-FILE-LINK for the Adabas Transaction Manager product library to all batch, TIAM and UTM tasks that will use Adabas Transaction Manager services. The Adabas Transaction Manager library and the Adabas System Coordinator library must be available to Adabas Transaction Manager tasks.



Note: As in previous releases, there is a requirement to use an unmodified ADALNK in certain areas. With Version 8.1 this restriction has been reduced to unmodified ADALNK being needed only when running standalone Adabas utility jobs.

12

Verifying the Installation

- Verify Basic Operation of the Adabas Transaction Manager Client Proxy 58
- Verify Communication with the Transaction Manager 58
- Verify Two-Phase Commit Operation 59

At the end of the installation process, you should verify that the installation was successful by performing the steps described in this section.

Verify Basic Operation of the Adabas Transaction Manager Client Proxy

▶ to verify basic operation of the transaction manager client proxy:

- 1 If it is currently executing, stop the TP system where you will use Adabas Transaction Manager Online Services.

This must be a TP system in which the Adabas Transaction Manager client proxy has been installed, with client runtime control setting `ATM=ON`.

- 2 Start (or restart) the TP system so that the Adabas Transaction Manager client proxy is loaded and initialized.
- 3 Log on to the Adabas Online System (AOS). Select Adabas Transaction Manager to invoke the Online Services application SYSATM.

SYSATM automatically searches for an active transaction manager in the local system. If the transaction manager is not active in the local system, a message is displayed indicating that status. You can ignore the message at this time.

- 4 Press `Enter`, then make a note of the Terminal ID that is displayed on the main menu.
- 5 Select Local Sessions, then select Local Sessions from the next menu.

One of the listed items should be recognizable either as the Terminal ID you noted in step 4 or the User ID of your current session, depending on your TP environment. This verifies the basic operation of the Adabas Transaction Manager client proxy.

Verify Communication with the Transaction Manager

▶ to verify communication with the transaction manager:

- 1 Start the transaction manager and re-enter the SYSATM application.

The transaction manager is identified automatically by the Adabas Transaction Manager client proxy when you enter SYSATM.

- 2 Select Transaction Manager Information, then Display Zap Information.

Displaying the list of applied fixes provided by this function verifies that communication with the transaction manager is functioning correctly, even if the list of fixes is empty.

- 3 Terminate your SYSATM session.

Verify Two-Phase Commit Operation

▶ to verify two-phase commit operation:

- 1 Start two databases with the parameter setting `ADARUN DTP=RM`: one database containing a standard Employees file and the other database containing a standard Vehicles file.
- 2 In library `SYSMTvrs` (where *vrs* is the version, revision, and system maintenance level of Adabas Transaction Manager), modify the test program DEMODTP so that it correctly refers to the Employees and Vehicles files.
- 3 Execute the DEMODTP program, supplying a non-zero value when prompted for a new personnel number.
- 4 When the message “About to commit” is displayed
 - start a new session in your TP system
 - log on to the Adabas Online System (AOS)
 - select Adabas Transaction Manager, then Transaction Manager Information, and then Active Transactions.

The display of your pending transaction should indicate that two databases are involved in it.



Note: This test is not valid in a CICS/RMI environment in which Natural executes in pseudo-conversational mode. In this mode, the INPUT statement that displays the message “About to commit” causes CICS to take a syncpoint at end-of-task and commit the pending Adabas updates. To verify correct operation in a CICS/RMI environment, this test should be done in a Natural session that was started with `PSEUDO=OFF`, or for which the runtime control `TransactionModel` has been set to DYNAMIC.



Note: This test is not valid in an IMS/TM environment in which RRMS acts as the superior transaction coordinator. In such an environment, the `ADARUN TMSYNCMGR` parameter should take the value NONE for the purposes of this test.

- 5 Display additional details of the transaction by marking it on the screen and pressing Enter.
- 6 Verify that the Adabas Transaction Manager client proxy’s view of the transaction is correct by returning to the main menu and selecting Local Sessions, then Local Sessions once again.
- 7 Return to the session where your transaction is waiting to be completed and press Enter.
- 8 Use SYSATM to check that the transaction is no longer open.

The client session itself should still be listed under Local Sessions, but it should no longer appear in the Transaction Manager Information “Active Transactions” list.

- 9 Execute the program DEMODTP again to ensure that the personnel number was changed consistently in the Employees and Vehicles files.

You can now be assured of global transaction integrity in client environments and databases where the software is installed.