

# Preparing for the Installation

This section of the documentation provides installation preparation information for Adabas Review in BS2000 operating system environments.

The installation of Software AG products is performed by installation *jobs*, which are either created *manually* or generated by System Maintenance Aid (SMA).

For each step of the installation procedure described below, the job number of a job performing the respective task is indicated. This job number refers to an installation job generated by SMA. If you are not using SMA, a sample installation job of the same number is provided in the job library on the Natural installation tape; you must adapt this example job to your requirements.

For information about using Software AG's System Maintenance Aid (SMA) for the installation process, refer to the *System Maintenance Aid Manual*.

This chapter covers the following topics:

- Prerequisites
  - Storage Requirements
  - About the Adabas Review Installation Tape
  - Installation Overview
- 

## Prerequisites

Other prerequisites for Adabas Review are described in *Software AG Product Support*.

## Operating System Level

Before installing this version of Adabas Review, ensure that you are running a currently-supported version of BS2000 OSD. For information on the BS2000 OSD versions supported by this release of Adabas Review, access Software AG's Empower web site at <https://empower.softwareag.com>. Log into Empower. Once you have logged in, you can expand Products in the left menu of the web page and select Product Roadmaps to access the Product Version Roadmaps application. This application allows you to review platform support information for specific Software AG products and releases.

## TP Monitor Support

This version of Adabas Review requires one of the following TP monitors:

- TIAM
- UTM

# Storage Requirements

Adabas Review must allocate storage to execute. Storage is required for

- the Adabas Review hub, if used;
- the REVIEW-BUFFER, used as a queueing area for Adabas command log records;
- reports that are executing;
- users accessing the database from the Adabas Review online system; and
- work areas used in Adabas Review's processing.

The type, purpose, and size of these storage areas is discussed in the following sections.

Adabas Review allocates storage *above the line* whenever it is permitted by the architecture of the machine and the operating system on which it is executing.

- Storage for the Hub
- Storage for the REVIEW-BUFFER
- Storage for Reports
- Storage for Online Users
- Storage for Work Areas

## Storage for the Hub

If you use Adabas Review in hub mode, the hub has a separate storage requirement for its operating queues and working areas. The queues are used to buffer the incoming command log records from the clients until the records can be sent to REVIEWB.

Two queues, both controlled by the database administrator (DBA), are used by the Adabas Review hub: the command queue (sized using the ADARUN parameter NC) and the attached buffer (sized using the ADARUN parameter NAB). For more information, read about these ADARUN parameters in *ADARUN Parameters*.

- Command Queue
- Attached Buffer

## Command Queue

The command queue stores information about the client nucleus such as job name, internal ID, etc. Each entry in the command queue represents one command log record from a client.

An entry exists for the time that a command log record is queued and awaiting selection from the hub until the time that the record is sent to REVIEWB. Once the command log record is sent to REVIEWB, the entry is released from the command queue.

This means that the command queue must be large enough to accommodate the backlog of command log records from the client nuclei. If the command queue is too small, it is possible that command log records will be dropped by the hub.

The ADARUN parameter that controls the command queue size is NC. The value of this parameter should be set higher for the hub than it is for individual client nuclei.

The NC value should be set to handle the arrival rate based on:

- the number of clients;
- their respective command processing limits;
- the processing power of the CPU(s); and
- the priority settings of the nuclei and hub address spaces, partitions, or regions.

### **Example**

If a client nucleus can process 2000 commands per second, then the expected arrival rate at the hub is a maximum of 2000 command log records per second. When possible, buffering occurs in the client nucleus to buffer several command log records and send them with one call to the hub.

There is no general rule for estimating the NC requirements for a particular hub. However, in this example, you could start with NC=1000 and monitor the results.

### **Attached Buffer**

The attached buffer is used to store the contents of the command log records and their associated data extensions.

As with the command queue, an element within the attached buffer is allocated to hold the command log record for the duration of time that the record is queued for selection, up to the time the record can be sent to REVIEWB. The element is freed once the record is sent to REVIEWB.

Also like the command queue, the attached buffer must be large enough to hold the queued command log records for the time required to stage the records for REVIEWB. Software AG recommends setting the parameter high to ensure that command log data is not dropped by the hub.

The ADARUN parameter controlling the attached buffer size is NAB. The value of this parameter should also be set higher for the hub than it is for individual client nuclei.

The NAB value must be large enough to buffer the data passed by the client nuclei. The amount of data passed by a client nucleus depends upon the Adabas Review report requirements (for example, whether control buffers are required or whether the I/O list option is being used).

### **Example**

The average size of a command log record and extensions, excluding control buffers, is 2500 bytes.

One approach would be to compute:

$$NAB = (NC * 2500 / 4096)$$

- where 4096 is the size of one NAB segment. If NC=1000 (see the example) , the starting value would be:

$$NAB = (1000 * 2500 / 4096) = 610$$

This computation assumes that there are no control buffers or I/O list elements being passed to the hub.

## Storage for the REVIEW-BUFFER

REVIEW-BUFFER is used to queue Adabas command log records to be sent to REVIEWB. In hub mode, it is located in the hub (server) address space.

The BUFFER-SEGMENTS parameter specifies the size of the REVIEW-BUFFER. Each buffer segment is 512 bytes. When a value for BUFFER-SEGMENTS is specified, the total storage indicated by the specification is split in half between 4K buffers and 32K buffers.

In addition, two INPUT statement parameters have been added as replacements for the BUFFER-SEGMENTS parameter. The BUFFER-SEGMENTS parameter will no longer be supported in a future release of Adabas Review. It is still valid in Adabas Review 4.5, but is ignored if the following parameters are specified:

Parameter	Description	Minimum Value	Default Value
BUFFERS-4K	The number of buffer pool entries that have a length of 4096 bytes or less. This parameter is usually specified along with the BUFFERS-32K parameter. If this parameter is not specified, any BUFFERS-32K parameter setting is ignored. If a value below the minimum value is specified for this parameter, the default is used.	124	256
BUFFERS-32K	The number of buffer pool entries that have a length of 4097 bytes or greater. This parameter is usually specified along with the BUFFERS-4K parameter. If this parameter is not specified, any BUFFERS-4K parameter setting is ignored. If a value below the minimum value is specified for this parameter, the default is used.	15	30

For more information about the interaction of the BUFFER-SEGMENTS, BUFFERS-4K, and BUFFERS-32K parameters, read *BUFFER-SEGMENTS Parameter Changes*

For z/OS, z/VSE, and BS2000, it is possible to execute with a REVIEW-BUFFER that is one megabyte.

A larger REVIEW-BUFFER provides a larger queueing area for command log records being sent to REVIEWB and decreases the possibility that Adabas will have to wait for REVIEWB to process these records in the event that REVIEW-BUFFER becomes full.

## Storage for Reports

### For Control Blocks

When a report is started, either using autostarted report definition parameters or by an online Adabas Review user, storage is allocated for control blocks that define the criteria for the collection of the data.

Typically, the storage allocation for control blocks is two (2) kilobytes, but may be as much as four (4) kilobytes if the report is a history report or the report specifies the collection of many fields.

### For Data Collection Areas

In addition to the report control blocks, storage is allocated for the collection of data. The data collection areas are allocated in two (2) kilobyte pieces and a subsequent data collection area is only allocated when the current area is full.

### Total Storage Limit

The total storage allocation for a report is limited by the MAXSTORE report parameter. When the total storage allocation for a report is equal to the MAXSTORE value, the report is marked as inactive and stops accumulating data. When a report is purged, all storage associated with the report is deallocated.

## Storage for Online Users

Adabas Review's online system uses Adabas calls to start, view, or purge a report. Each request requires that Adabas Review perform some processing to fulfill the request.

- Each request from the Adabas Review online system results in the allocation of a piece of storage (about 500 bytes or one-half kilobyte) that is deallocated when the request has been satisfied.
- To maintain the integrity of each request, Adabas Review allocates an area for each user requesting Adabas Review to service a request. For example, a request to view a report requires an Adabas call for each record that is to be viewed online.
- In cases where more than one user is viewing the same or different reports, Adabas Review must remember the status of each user between Adabas calls.

## Storage for Work Areas

Adabas Review allocates storage for work areas and areas used for reading from and writing to files. These areas are typically small and are kept and used throughout the time that Adabas Review is active.

## About the Adabas Review Installation Tape

The Adabas Review installation tape is a standard label tape with the volume serial number REV $vrs$ , where the notation  $vrs$  represents the version, revision, and SM level of the product.

- Copying the Installation Tape When Not Using SMA
- Copying the Installation Tape Using SMA

- Installation Tape Contents
- Source Library Members

## Copying the Installation Tape When Not Using SMA

### Note:

If you are using SMA, refer to the *System Maintenance Aid* documentation. If you are not using SMA, follow the instructions below.

This section explains how to copy all files from tape to disk. You will then need to perform the individual installation procedure for each component to be installed.

- Step 1: Copy the Library SRVvrs.LIB from Tape to Disk
- Step 2: Copy the Procedure COPY.PROC from Tape to Disk
- Step 3: Copy all Product Files from Tape to Disk

### Step 1: Copy the Library SRVvrs.LIB from Tape to Disk

This step is not necessary if you have already copied the library SRVvrs.LIB from another Software AG installation tape. For further information, refer to the element #READ-ME in this library. The library SRVvrs.LIB is stored on the tape as a sequential file named SRVvrs.LIBS containing LMS commands. The current version vrs 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=SRVvrs.LIBS,-
/  VOLUME=volser, DEV-TYPE=tape-device)
/ADD-FILE-LINK LINK-NAME=EDTSAM, FILE-NAME=SRVvrs.LIBS,-
/  SUPPORT=*TAPE(FILE-SEQ=3), 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=SRVvrs.LIBS'
@WRITE  'SRVvrs.LIBS'
@HALT
/ASS-SYSDTA  SRVvrs.LIBS
/MOD-JOB-SW  ON=1
/START-PROG  $LMS
/MOD-JOB-SW  OFF=1
/ASS-SYSDTA  *PRIMARY
```

where:

*tape-device* is the device type of the tape, for example: TAPE-C4  
*volser* is the VOLSER of the tape (see *Report of Tape Creation*)

### Step 2: Copy the Procedure COPY.PROC from Tape to Disk

To copy the procedure COPY.PROC to disk, call the procedure P.COPYTAPE in the library SRVvrs.LIB:

```
/CALL-PROCEDURE  (SRVvrs.LIB,P.COPYTAPE), -
/  (VSNT=volser, DEVT=tape-device)
```

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

### Step 3: Copy all Product Files from Tape to Disk

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

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

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

## Copying the Installation Tape Using SMA

If you are installing Adabas Review using the Software AG System Maintenance Aid (SMA), refer to the *System Maintenance Aid* documentation and to the information provided with the installation tape for specific installation instructions.

### Installation Tape Contents

The installation tape contains the files listed below. The sequence of the files is shown in the *Report of Tape Creation* delivered with the installation tape.

File Name	Contents
REV <sub>vrs</sub> .INPL	Adabas Review Natural objects
REV <sub>vrs</sub> .MOD	Adabas Review module library
REV <sub>vrs</sub> .SRC	Adabas Review source modules and installation jobs
REV <sub>vrs</sub> .SYSF	Empty Adabas Review repository file
REV <sub>vrs</sub> .ZAPS	Zaps required for Adabas Review on this platform

**Note:**

Throughout this documentation, the notation *vrs* stands for the version, revision, and system maintenance level of the Adabas Review being installed.

### Source Library Members

Source library members for Adabas Review are as follows:

**Note:**

Type "J" indicates a job member, "S" indicates a source member, and "M" indicates a macro member.

Member	Type	Description
ASMBS2	J	Sample job for macro assembly.
EXPAND	J	Sample job to upgrade an Adabas Review 3.4.4 repository to an Adabas Review 4.1 repository.
EXPAND1	J	Sample job to upgrade an Adabas Review 4.1 repository to an Adabas Review 4.2 repository.
EXPAND2	J	Sample job to upgrade an Adabas Review 4.2 repository to an Adabas Review 4.3 repository.
EXPAND3	J	Sample job to upgrade an Adabas Review 4.3 or 4.4 repository to an Adabas Review 4.5.1 repository.
EXPAND4	J	Sample job to upgrade an Adabas Review 4.5.1 repository to an Adabas Review 4.5.2 repository.
HUBJCL	J	Sample job to start the Adabas Review hub.
LOCAL	J	Sample job to start the Adabas Review in local mode.
P.ADALNK	J	Sample job to link the Adabas Review user exit with ADALNK.
P.GENERATE	J	Sample job to allocate and generate the files used by Adabas Review.
P.REVBATCH	J	Sample JCL to process a sequential command log by the batch component of Adabas Review.
REGEQU	M	Used by REVUEX5.
REVUEX1	S	Sample user exit 1 (User field exit) program.
REVUEX5	S	Sample user exit 5 program.
REVUXLOG	S	Sample Assembler source code for an Adabas Review command logging user exit.
REVUXSUM	S	Sample summary report user exit.
UEX5PARAM	M	Used by REVUEX5.
WTO	M	Used by REVUEX5.
ZAPOPT	S	Contains optional zaps for installing Adabas Review.

## Installation Overview

Adabas Review is installed in two phases:

1. Phase 1: Install non-TP-specific components. All steps in this phase are identical regardless of the TP monitor in use:
  - Install Adabas Review under Natural;
  - Install the Adabas Review repository;



- Install Adabas Review under Adabas;
  - Install the Adabas Review hub (hub mode only);
  - Optional installation procedures.
2. Phase 2: Install TP-specific components. Separate procedures are used to install Adabas Review under each of the supported TP monitors: TIAM, batch, and UTM.

Phase one procedures are described in *Installation: Phase 1*; phase two procedures in *Installation: Phase 2*.