

Adabas Review

Installation and Operations for z/VSE

Version 4.5.1

June 2014

This document applies to Adabas Review Version 4.5.1.

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

Copyright © 2014 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors..

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

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://documentation.softwareag.com/legal/>.

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

This software may include portions of third-party products. For third-party copyright notices and license terms, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". This document is part of the product documentation, located at <http://documentation.softwareag.com/legal/> and/or in the root installation directory of the licensed product(s).

Document ID: REV-VSE-INSTALL-451-20140626

Table of Contents

Installation and Operations for z/VSE	v
1 Conventions	1
2 Preparing for the Installation	3
Prerequisites	4
Storage Requirements	4
The Adabas Review Installation Tape	9
Installation Overview	11
3 Installation: Phase 1	13
Copying the Tape Contents to Disk	14
Install Adabas Review under Natural	15
Install the Adabas Review Repository	17
Install Adabas Review under Adabas	18
Install the Adabas Review Hub	23
Optional Installation Procedures	24
4 Installation: Phase 2	25
Install Adabas Review under Com-plete	26
Install Adabas Review under CICS	26
Install Adabas Review under Batch	27
5 Starting Adabas Review	29
Starting Adabas Review for the First Time	30
Accessing Adabas Review	34
Verifying the Installation	35
6 Operations	39
Assigning Logical Units	40
Processing ABENDs	41
Adding Adabas Startup Statements at Installation	42
Files Used by Adabas Review	43
Editing the RVUEXI Parameter File	47
Command Logging Considerations	48
Summary Logging Considerations	49
Modifying Configuration Parameters	51
User Exit 5 (Adabas Review Hub Event Handler)	53
Adabas Review Natural User Exits	55
7 ADARUN Parameters for Adabas Review in z/VSE	57
ADARUN Parameter Syntax	58
PROGRAM : Program to Run	59
REVIEW : Adabas Review Control	60
CT : Command Timeout Limit	61
FORCE : Allow Nucleus Database ID or Review Hub Table Entry Overwrite	62
LOCAL : Local Nucleus or Adabas Review Hub	63
NAB : Number of Attached Buffers	64
NC : Number of Command Queue Elements	65
REVFILTER : Review Record Filtering Control	66

REVLOGBMAX: Logged Buffer Size Limit for Review	66
REVLOGMAX: Total Logged Buffer Size Limit for a Review Command	67
SVC : SVC Number	67
8 Operator Commands	69
Entering Operator Commands	70
Operator Command Overview	70
Index	73

Installation and Operations for z/VSE

Adabas Review is a set of tools for monitoring the performance of Adabas environments and the applications executing within them. Information retrieved about Adabas usage helps you tune application programs to achieve maximum performance with minimal resources.

This part of the Adabas Review documentation provides information for installing and maintaining Adabas Review

- in local mode in the Adabas address space; or
- in hub mode as a hub (server) in its own address space with only interface (client) modules in the Adabas address space.

The Adabas Review installation documentation for z/VSE is organized as follows:

<i>Preparing for the Installation</i>	Provides the information needed to prepare for the installation of Adabas Review under z/VSE operating system environments under the teleprocessing monitors Com-plete or CICS.
<i>Installation: Phase 1</i>	Provides instructions for installing the components of Adabas Review that are independent of any particular TP monitor (phase 1).
<i>Installation: Phase 2</i>	Provides instructions for installing the components of Adabas Review that depend on a particular TP monitor (phase 2).
<i>Starting Adabas Review</i>	Discusses the initialization procedures performed by Adabas Review during startup.
<i>Operations</i>	Describes operation procedures for Adabas Review.
<i>ADARUN Parameters for Adabas Review in z/VSE</i>	Discusses the Adabas ADARUN initialization parameters, most for use with the hub only, in z/VSE environments.
<i>Operator Commands</i>	Describes operator commands for use with the hub only.

1 Conventions

In the product documentation, the notation *vrs*, *vr*, or simply *v* is often used as a placeholder for the current product version, for example, in data set or module names.

Placeholder	Meaning	Definition
<i>v</i>	version	Major Version The first digit of the product version number indicates major architecture and functionality implementation or enhancement that adds value to the product.
<i>r</i>	release	Minor Version The second digit of the version number indicates functionality addition or enhancement that adds value to the product.
<i>s</i>	system maintenance level	Correction Level Correction levels contain error corrections only, without new functionality, including documentation of all modifications and repairs. In case it is necessary to include functional changes into a correction level, an exception handling process ensures that corresponding quality assurance activities are triggered. These functional changes are documented. The main target is to avoid impacts when you install such a correction level. The third number of an Adabas version denotes the system maintenance level. On certain platforms supported by Adabas, additional levels may exist, such as update package, patch level, service pack and hot fix.

2 Preparing for the Installation

- Prerequisites 4
- Storage Requirements 4
- The Adabas Review Installation Tape 9
- Installation Overview 11

This section of the documentation provides installation preparation information for Adabas Review under z/VSE operating system environments.

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

Prerequisites

Other prerequisites for Adabas Review are described in *Software AG Product Support*, in the *Adabas Review Release Notes*.

TP Monitor Support

This version of Adabas Review supports the following TP monitors:

- Com-plete
- CICS/VSE or CICS/TS

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](#)

- [Size of Adabas Review in z/VSE Environments \(Local Mode Only\)](#)

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*, in the *Adabas Review Concepts Manual*.

- [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 <code>BUFFERS-32K</code> parameter. If this parameter is not specified, any <code>BUFFERS-32K</code> 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 <code>BUFFERS-4K</code> parameter. If this parameter is not specified, any <code>BUFFERS-4K</code> 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*, in the *Adabas Review Release Notes*

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.

Size of Adabas Review in z/VSE Environments (Local Mode Only)

In VSE environments, the available GETVIS in the Adabas partition may need to be increased to accommodate Adabas Review. An increase of 500K is normal.

The Adabas Review Installation Tape

This section describes the contents of the Adabas Review installation tape and the space requirements for each data set. Sample JCS to accomplish the transfer is also included.

Refer to the "Report of Tape Creation" for the volume serial number, density, media type, data set names, and data set sequence numbers for the SM level being installed.

- [Installation Tape Description](#)
- [Disk Space Requirements](#)
- [Sublibrary Members](#)

Installation Tape Description

The installation tape contains the following data sets:

Data Set	Created Using	Contents
REVvrs.INPL	NATUNLD	Adabas Review Natural objects
REVvrs.SYSF	ADAULD	Adabas Review repository file
REVvrs.VSEZAPS	--	Adabas Review ZAP data set
REVvrs.VSELIBR	LIBR BACKUP	The Adabas Review sublibrary; contains relocatable objects, phases, source, and example installation jobs

Disk Space Requirements

The space requirements for each of the data sets on the installation tape is shown below:

Data Set Name	Cylinders (3390)
INPL	8
SYSF	1
VSEZAPS	1
VSELIBR	4

Additionally, the alternate history file, which is created when installing Adabas Review under Adabas, requires additional space as follows:

Data Set Type	Cylinders (3390)
Alternate history file	2

Sublibrary Members

The Adabas Review sublibrary members are listed below. The members are listed by type, where

- "A" indicates source (for example, Assembler user exit samples, macros, etc.)
- "X" indicates job control statements or job streams.

Members of type A are:

Member	Description
LOGUEXIT	Sample Assembler source code for an Adabas Review command logging user exit.
RAOSLUBS	Sample Assembler code to change the default logical units used by Adabas Review.
REVCOST	Sample Adabas Review parameters used to produce an Adabas cost accounting report.
SAMPUEX5	Sample user exit 5 program.
SUMEXIT	Sample summary report user exit.
UEX5PARAM	Adabas user exit 5 macro used for the Adabas Review hub.
ZAPOPT	Adabas Review optional zaps.

Members of type X are:

Member	Description
ARCHIVE	Sample JCS to define Adabas Review to MSHP.
BATCHRPT	Sample job to create batch reports.
DBFILES	Create Adabas Review data sets.
EXPAND3	Sample JCS to upgrade a version 4.3 or 4.4 repository to version 4.5.
HISTCOMP	Sample JCS to compress history data from a batch Natural execution.
HISTDEL	Sample JCS to delete history data from a batch Natural execution.
HISTVIEW	Sample JCS to view history data from a batch Natural execution.
HUBJCS	Sample JCS to start the Adabas Review hub server.
LINKREV	Sample job to relink Adabas Review after applying maintenance.
LNKLUBS	Sample JCS to assemble and link RAOSLUBS system file number assign module.
LOCJCS	Sample JCS to run Adabas Review local nucleus.
LREVCIC	For CICS installations, sample JCS to assemble the CICSGBLS module and link it with the necessary Adabas LNK and Adabas Review exit objects.
LREVLCO	For Com-plete installations, sample JCS to assemble the LCOGBL module and link it with the necessary Adabas LNK and Adabas Review exit objects.
LREVLNK	For batch installations, sample JCS to assemble the LNKGBLS module and link it with the necessary Adabas LNK and Adabas Review exit objects.
LREVLNKR	For batch installations, sample JCS to assemble the LNKGBLS module and link it with the necessary reentrant Adabas LNK and Adabas Review exit objects.
REVCLCOP	Sample JCS to copy and set end-of-file for a sequential command log created by Adabas Review.
REVIEWB	Sample JCS to process a sequential command log by the batch component of Adabas Review.
REVINPL	Sample JCS to INPL the Adabas Review programs and DDMs from the INPL data set to the Natural system files.
REVLOAD	Sample JCS to load the Adabas Review repository file into an Adabas environment.
REVPROC	Sample job to catalog the Adabas Review standard label procedure.

Installation Overview

Adabas Review is installed in two phases:

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. Install TP-specific components. Separate procedures are used to install Adabas Review under each of the supported TP monitors: Com-plete and CICS.

Phase one procedures are described in section *Installation (Phase 1)*; phase two procedures in section *Installation (Phase 2)*.

3 Installation: Phase 1

- Copying the Tape Contents to Disk 14
- Install Adabas Review under Natural 15
- Install the Adabas Review Repository 17
- Install Adabas Review under Adabas 18
- Install the Adabas Review Hub 23
- Optional Installation Procedures 24

Adabas Review can be installed in local mode in the Adabas address space, or a hub (server) located in its own address space with an interface (client) located in the address space of the Adabas being monitored. The procedures are the same except as noted.

Adabas Review is installed in two phases: this section of the documentation describes phase one, which comprises all the steps that are independent of any particular TP monitor environment.

To complete phase 1 of the installation, include all steps that apply to the type of installation you have chosen and follow them in order.

Phase 2, which comprises the steps that are specific to the TP monitor being used, is described in section [Installation \(Phase 2\)](#).

Copying the Tape Contents to Disk

Use the procedure described in this section to allocate and copy the Adabas Review installation files to disk. All data set sizes are based on 3380 device types.

Step 1: Copy Data Set COPYTAPE.JOB From Tape To Disk

The data set COPYTAPE.JOB (file 5) contains the JCL to unload all other existing data sets from tape to disk. To unload COPYTAPE.JOB, use the following sample JCL:

```
* $$ JOB JNM=LIBRCAT,CLASS=0, +
* $$ DISP=D,LDEST=(*,UID),SYSID=1
* $$ LST CLASS=A,DISP=D
// JOB LIBRCAT
* *****
* CATALOG COPYTAPE.JOB TO LIBRARY
* *****
// ASSGN SYS004,nnn <----- tape address
// MTC REW,SYS004
// MTC FSF,SYS004,4
ASSGN SYSIPT,SYS004
// TLBL IJSYSIN,'COPYTAPE.JOB'
// EXEC LIBR,PARM='MSHP; ACC S=lib.sublib' <----- for catalog
/*
// MTC REW,SYS004
ASSGN SYSIPT,FEC
/*
/&
* $$ EOJ
--- where
  nnn is the tape address, and lib.sublib is the library and sublibrary of the catalog.
```

Step 2: Modify COPYTAPE.JOB

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

Supply values for the following user substitutions in the sample JCS provided in later steps in this procedure.

<i>vrs</i>	with the version, revision, and system maintenance level of Adabas Review
<i>nnnn</i>	with the starting track number;
<i>vvvvvvv</i>	with the tape volume serial number;
<i>ttt</i>	with the tape unit address;
<i>f</i>	with the number of files to forward space; refer to the <i>Report of Tape Creation</i> for this information.

Step 3: Submit COPYTAPE.JOB

Submit COPYTAPE.JOB to unload all other data sets from the tape to your disk.

Install Adabas Review under Natural

This section applies to both local and hub mode installations. It includes the following steps:

- [Step 1. INPL Adabas Review](#)
- [Step 2. Create a Natural profile using the SYSPARM facility](#)
- [Step 3. Define Adabas Review to Natural Security](#)

Step 1. INPL Adabas Review



Note: There is no longer a SYSREV or SYSREVUP library. The entire application is now contained in the SYSREVDDB library.

INPL the Adabas Review programs and DDMs from the INPL data set to your Natural system files.

You may use any of your site-dependent Natural INPL JCS. Sample job *REVINPL.X* is provided in the Adabas Review library.

Step 2. Create a Natural profile using the SYSPARM facility

► To create a Natural profile using the SYSPARM facility:

- 1 Include the following parameter settings in the Natural profile:

Parameter	Requirement
LS=250	minimum
PS=80	minimum
MADIO=5000	minimum
MAXCL=0	minimum
ESIZE=40	minimum
ADAPRM=ON	required in order to for Adabas Review to report on Natural information.
NETWORK(7),AM=PC	required for PC downloads. NETWORK is a Natural macro used to define the work file(s) to be used; 'AM' is the access method. For more information, see the Natural documentation.
NTPRINT(1)	Adabas Review uses Natural printer 1 to generate report display programs in the Natural source area. For more information, see the Natural documentation.

- 2 Add a Natural NTFILE definition for the physical database ID and file number of the Adabas Review repository file as follows:

```
NTFILE 241,dbid,fnr
```

Replace *dbid* and *fnr* with the database ID and file number, respectively, of the Adabas Review repository.



Note: The Adabas Review repository may not reside on a database with a database ID (DBID) of 255. If the database ID is 255, Adabas Review cannot be accessed in local or hub mode. However, a database with a DBID of 255 can send data to a hub.

- 3 Reassemble and link the NATPARM module to your Natural nucleus.

Step 3. Define Adabas Review to Natural Security



Note: If the Adabas Review application SYSREVDDB is made private (i.e., the parameter PEOPLE=Y is specified), each user of Adabas Review must be linked to the SYSREVDDB application.

▶ If Natural Security is installed, complete the following steps:

- 1 Define the SYSREVDDB library for the Adabas Review system.
- 2 Define the following Adabas Review DDMs to Natural Security as public DDMs:

```
REVIEW-Adabas-V431-CLOG
REVIEW-FNAT
REVIEW-Adabas-V431-SYSTEM
```

Install the Adabas Review Repository

This procedure applies to both local and hub mode installations.

The Adabas Review repository is a system file used for storing descriptions of interactive reports, target definitions, and for saving historical data accumulated by Adabas Review reports. Any Adabas file may be used to contain the Adabas Review repository. The corresponding file number should also be reflected in the NATPARMs used to invoke Adabas Review.

Hub mode only: The Adabas Review repository can be created on any database accessible to Natural. It does not need to be on a database that is monitored by the Adabas Review hub.



Notes:

1. The Adabas Review repository may not reside on a database with a database ID (DBID) of 255. If the database ID is 255, Adabas Review cannot be accessed in local or hub mode. However, a database with a DBID of 255 can send data to a hub.
2. To convert an Adabas Review version 4.3 or 4.4 repository to a version 4.5 repository, tailor and run the EXPAND3 job from the sample jobs library.

This section covers the following topics:

- [Step 1. Modify the JCS for loading the Adabas Review repository](#)
- [Step 2. Load the Adabas Review repository file](#)
- [Step 3. Convert the repository file](#)

- [Step 4. Convert your history data](#)

Step 1. Modify the JCS for loading the Adabas Review repository

Before submitting the job REVLOAD.X provided in the Adabas Review sublibrary, change

- the DATABASE=*dbid* parameter in the two ADARUN statements to reflect the DBID number of the database that will contain this file;



Note: The Adabas Review repository may not reside on a database with a database ID (DBID) of 255. If the database ID is 255, Adabas Review cannot be accessed in local or hub mode. However, a database with a DBID of 255 can send data to a hub.

- the SVC=*svc* parameter of the same two ADARUN statements to reflect the number of the SVC used for the database defined in the above step; and
- the ADALOD LOAD FILE=*fnr* statement to reflect the number of the file that will contain the Adabas Review file.

Step 2. Load the Adabas Review repository file

Load the Adabas Review repository file using the job REVLOAD.X as modified in step 1.

Step 3. Convert the repository file

If you used the same repository file for a version of Adabas Review older than this release, use the EXPAND3 job in the JOBS library to convert the repository.

Step 4. Convert your history data

If you used the same repository file for a version of Adabas Review older than this release, Adabas Review 4.5 requires that you convert your history data prior to running any new reports in 4.5. For complete information on doing this, read *Migration from Previous Versions*, in the *Adabas Review Release Notes*.

Install Adabas Review under Adabas

To install Adabas Review under Adabas, complete the following steps:

- [Step 1. Create Adabas Review-specific data sets](#)
- [Step 2. Modify the Adabas initialization parameters](#)
- [Step 3: Apply the required ZAPs](#)
- [Step 4: Modify the Adabas Startup JCS](#)

- [Step 5: Define Adabas Review to MSHP](#)

Step 1. Create Adabas Review-specific data sets

Use the sample JCS member DBFILES.X in the Adabas Review sublibrary to create the data sets necessary to store autostarted report definitions, parameters, and the alternate history file.

▶ To create Adabas Review-specific data sets:

- 1 Modify DBFILES.X by replacing

s	with the current system maintenance level of Adabas Review
nnnn	with the starting track number for each step in the job
vvvvvv	with the tape volume serial number

- 2 Execute the modified DBFILES.X.

Step 2. Modify the Adabas initialization parameters

See the section [ADARUN Parameters](#) for information about the relevant ADARUN parameters.

Modify your existing Adabas ADARUN parameters to include the following:

- local mode only:

```
ADARUN REVIEW=LOCAL
```

- hub mode only for each database to be monitored by Adabas Review:

```
ADARUN REVIEW=hubid
ADARUN UEX5=user-exit (optional)
```

where *hubid* is the Adabas Review hub ID and UEX5 is optional (see section [User Exit 5 \(Adabas Review Hub Event Handler\)](#)).

Step 3: Apply the required ZAPs

Step 4: Modify the Adabas Startup JCS

This section describes how to modify the Adabas startup JCS for local mode and for hub mode.

Local Mode

▶ **To modify the Adabas startup JCS for local mode installations:**

- 1 Before you modify the Adabas startup JCS, you may need to increase the available GETVIS in the Adabas partition to accommodate Adabas Review. An increase of 500K is normal.

RVUALT Considerations

- RVUALT refers to the data set that contains the alternate history file. You may use the same data set you used for earlier versions.
- If Adabas Review is installed on multiple databases, a RVUALT data set must be allocated for each database.
- Refer to section *RVUALT History File* in *Operations* for RVUALT guidelines.

RVUAUT1 and RVUAUT2 Considerations

- RVUAUT1 and RVUAUT2 refer to the data sets that contain the report definitions for autostarted reports.
- You can use the distributed RVUAUT1 and RVUAUT2 files from earlier versions.



Note: In a future release of Adabas Review, RVUAUT2 will no longer be supported. RVUAUT1 only will be used for autostarted reports.

To modify the startup JCS for local mode

1. Add the following statements to the Adabas startup JCS replacing

vrs	with the current version, revision, and system maintenance level of Adabas Review
vvvvvv	with the tape volume serial number



Note: In Adabas Review, each logical unit (SYS number) is assigned to a specific Adabas Review function (see *Assigning Logical Units*, elsewhere in this guide). If these logical units conflict with the assignments made in the Adabas data container procedure or the nucleus startup JCS, we recommend you change the SYS number allocation for Adabas to a number range above 50 in a copied procedure for use by the Adabas Review Local Nucleus. The currently assigned logical units are as shown below:

```

// ASSGN SYS005,IGN                RVUPARM - batch parameters
// ASSGN SYS007,DISK,VOL=vvvvvv,SHR  RVUAUT1, RVUAUT2, RVUFLD
// ASSGN SYS008,DISK,VOL=vvvvvv,SHR  RVUALT - alt history file
// ASSGN SYS020,SYSLST             RVUPRT0 - printer
// ASSGN SYS021,SYSLST             RVUPRT1 - printer
// ASSGN SYS022,SYSLST             RVUPRT2 - printer
// ASSGN SYS023,SYSLST             RVUPRT3 - printer
// ASSGN SYS041,DISK,VOL=vvvvvv,SHR  RVUCARD - GENCARD
// ASSGN SYS006,DISK,VOL=vvvvvv,SHR  RVUEXI - operating env parameters
// ASSGN SYS019,SYSLST             RVUEXP - EXI parameters printer
*
// DLBL RVUAUT1,'REVvrs.AUT01',0
// EXTENT SYS007,vvvvvv,1,0,nnnn,5
// DLBL RVUAUT2,'REVvrs.AUT02',0
// EXTENT SYS007,vvvvvv,1,0,nnnn,5
*
// DLBL RVUCARD,'REVvrs.CARD.FILE',0
// EXTENT SYS041,vvvvvv,1,0,nnnn,5
*
// DLBL RVUALT,'REVvrs.ALT.FILE',0
// EXTENT SYS008,vvvvvv,1,0,nnnn,30
*
// DLBL RVUEXI,'REVvrs.EXI.PARAMETERS',0
// EXTENT SYS006,vvvvvv,1,0,nnnn,5
*
// DLBL RVUFLD,'REVvrs.USER.FIELDS',0
// EXTENT SYS007,vvvvvv

```

2 Add the following statements to the Adabas startup JCS, replacing

<i>vrs</i>	with the current version, revision, and system maintenance level of Adabas Review
<i>vvvvvv</i>	with the tape volume serial number



Note: In Adabas Review, each logical unit (SYS number) is assigned to a specific Adabas Review function (see [Assigning Logical Units](#), elsewhere in this guide). If these logical units conflict with the assignments made in the Adabas data container procedure or the nucleus startup JCS, we recommend you change the SYS number allocation for Adabas to a number range above 50 in a copied procedure for use by the Adabas Review Local Nucleus. The currently assigned logical units are as shown below:

```

// ASSGN SYS005,IGN                RVUPARM - batch parameters
// ASSGN SYS007,DISK,VOL=vvvvvv,SHR  RVUAUT1, RVUAUT2, RVUFLD
// ASSGN SYS008,DISK,VOL=vvvvvv,SHR  RVUALT - alt history file
// ASSGN SYS020,SYSLST              RVUPRT0 - printer
// ASSGN SYS021,SYSLST              RVUPRT1 - printer
// ASSGN SYS022,SYSLST              RVUPRT2 - printer
// ASSGN SYS023,SYSLST              RVUPRT3 - printer
// ASSGN SYS041,DISK,VOL=vvvvvv,SHR  RVUCARD - GENCARD
// ASSGN SYS006,DISK,VOL=vvvvvv,SHR  RVUEXI - operating env parameters
// ASSGN SYS019,SYSLST              RVUEXP - EXI parameters printer
*
// DLBL RVUAUT1,'REVvrs.AUT01',0
// EXTENT SYS007,vvvvvv,1,0,nnnn,5
// DLBL RVUAUT2,'REVvrs.AUT02',0
// EXTENT SYS007,vvvvvv,1,0,nnnn,5
*
// DLBL RVUCARD,'REVvrs.CARD.FILE',0
// EXTENT SYS041,vvvvvv,1,0,nnnn,5
*
// DLBL RVUALT,'REVvrs.ALT.FILE',0
// EXTENT SYS008,vvvvvv,1,0,nnnn,30
*
// DLBL RVUEXI,'REVvrs.EXI.PARAMETERS',0
// EXTENT SYS006,vvvvvv,1,0,nnnn,5
*
// DLBL RVUFLD,'REVvrs.USER.FIELDS',0
// EXTENT SYS007,vvvvvv

```

- 3 Modify the Adabas startup JCS to execute the Adabas Review procedure and add the Adabas Review sublibrary to the LIBDEF search chain.

Hub Mode: Interface Install

▶ To install the client interface in hub mode:

- Repeat the following substeps for each database to be monitored.

Each database to be monitored must use the same SVC as the Adabas Review hub.

If the Adabas Adarun parameter LOGCLEX=Y is specified to write the Adabas Review specific Command Log Extension (CLEX) to the command log, the Adabas Review library must be available in the client address space. If Adabas Review modules are not available, the warning message ARVU20 is issued.

Step 5: Define Adabas Review to MSHP

Maintenance fixes are distributed in MSHP format. In order to apply these maintenance fixes, Adabas Review must be defined to MSHP as a product/component using the MSHP ARCHIVE process.

Modify and run the sample job stream ARCHIVE.X in the Adabas Review sublibrary to define Adabas Review to MSHP. Make sure you have specified the Adabas Review and the Adabas library in your MSHP job which applies the fixes for Adabas Review. For example:

```
// LIBDEF OBJ,SEARCH=(SMALIB.REVvrs,SMALIB.ADAvrs),TEMP
```

Install the Adabas Review Hub

Complete the steps in this section to install the Adabas Review Hub.

- [Step 1. Modify the sample JCS member HUBJCS.X](#)
- [Step 2. Start the HUBJCS job](#)

Step 1. Modify the sample JCS member HUBJCS.X

▶ **When modifying the sample JCS member, be sure to:**

- 1 Correct any library names or file names.
- 2 Modify the ADARUN parameters:
 - Modify the ADARUN parameter `REVIEW` to reflect the TARGET ID you wish to use for the Adabas Review hub.
 - Modify the ADARUN parameter `SVC` to reflect the correct SVC number. This SVC must be the same as that used by the sending Adabas nuclei.

The ADARUN parameters supplied in the sample JCL member HUBJCS.X are the only ones recognized for setting up the hub nucleus. See section [ADARUN Parameters](#) for information about these ADARUN parameters.



Note: Software AG recommends that the dispatching priority of the Adabas Review hub be set higher than that of the sending Adabas nuclei.

- 3 Verify that the RVUALT DLBL statement refers to the data set that contains the alternate history file.

The RVUALT or "alternate history" file is used to contain history information if Adabas Review is unable to access the Adabas Review repository. You may use the same RVUALT data set you used for earlier versions.

Refer to section [Operations](#) for RVUALT guidelines.

- 4 Verify that the RVUAUT1 and RVUAUT2 DLBL statements refer to the data sets that contain the report definitions for autostarted reports.

You can use the distributed RVUAUT1 and RVUAUT2 files from earlier versions.



Note: The reports contained in the RVUAUTn files have a TARGET= parameter that specifies the database that the report will monitor. The default setting is TARGET=001. You may wish to change this setting.

The RVUFLD DLBL statement refers to the data set that contains the user-defined field parameters.

Step 2. Start the HUBJCS job

Run the HUBJCS member you created in the previous step.

Optional Installation Procedures

Implement Support for Adabas Native SQL

From within Adabas Native SQL, use the global ADACALL statement with the LAST parameter to specify that the Adabas call will use the seventh parameter.

Refer to the *Adabas Native SQL Documentation*.

Implement Support for Reporting from Batch Natural

This (optional) step applies both to local and hub mode installations.

▶ To report on Natural activity from a batch Natural job:

- Modify and relink the Adabas link routine for batch/TSO as described in [Install Adabas Review under Batch](#), elsewhere in this guide.

4 Installation: Phase 2

- Install Adabas Review under Com-plete 26
- Install Adabas Review under CICS 26
- Install Adabas Review under Batch 27

The second phase installs the components that are specific to the particular TP monitor in use at your site. Separate procedures are used to install Adabas Review under each supported TP monitor: Com-plete, CICS, and batch.

Install Adabas Review under Com-plete

▶ To install the Com-plete components of Adabas Review for use with the Adabas 8 Com-plete link routine:

- 1 Use the JCS contained in the sublibrary member LREVLCO.X to set and assemble the LCOGBL module and link this to make the ADALCO phase with the Adabas Review link routine exit.



Note: The target of the link in this step must be a sublibrary in the Com-plete LIBDEF PHASE search concatenation.

- 2 Restart Adabas and Com-plete.
- 3 Initialize and test Adabas Review.

Refer to section [Starting Adabas Review](#) for information about initializing and verifying the installation of the online portion of Adabas Review.

Install Adabas Review under CICS

▶ To install the CICS components of Adabas Review for use with the Adabas 8 CICS link routine:

- 1 Modify the sample job LREVLCIC.X to set the SVC, DBID, library, and sublibrary information and submit it.
- 2 Restart Adabas and CICS.
- 3 Initialize and test Adabas Review.

Refer to section [Starting Adabas Review](#) for more information about initializing and verifying the installation of the online portion of Adabas Review.

Install Adabas Review under Batch

This section provides instructions for installing the batch components of Adabas Review with the Adabas batch link routines.

▶ **To install the batch components of Adabas Review for the Adabas 8 batch link routine:**

- 1 Edit member LNKGBLS.A in the Adabas distribution sublibrary. Provide values for the LOGID, SVC, GBLNAME, and other keywords to suit your installation requirements. Be sure to set the LGBLSET keywords:

```
REVIEW=YES  
OPSYS=VSE
```



Note: If you elect to use the reentrant link routine, edit member LNKRGBL.A in the Adabas distribution sublibrary.

- 2 Edit the ALNKLNK8.X member found in the Adabas 8 sublibrary. This member will assemble and catalog the LNKGBLS.A module and link it and any desired exits with the LNKVSE8.OBJ module to create the ADALNK.PHASE member for Adabas 8. The ALNKLNK8.X member includes sample link-edit control cards to support UES by including the LNKUES.OBJ. module with the ASC2EBC and EBC2ASC translation tables. Modify the link-edit control cards to include any additional Software AG exit or user exit, as specified in the updated LNKGBLS.A member.



Note: If you elect to use the reentrant link routine, edit member ALNKLNR8.X in the Adabas distribution sublibrary.

- 3 Provide the ADALNK.PHASE member in the LIBDEF search chain for the job step that will require Adabas database access or Software AG services.



Note: If you elect to use the reentrant link routine, provide the ADALNKR.PHASE member in the LIBDEF search chain for the job step that will required Adabas database access or Software AG services.

5 Starting Adabas Review

- Starting Adabas Review for the First Time 30
- Accessing Adabas Review 34
- Verifying the Installation 35

This section of the documentation describes the procedures for setting up and using Adabas Review after the installation has been completed.

Starting Adabas Review for the First Time


If this is the first time you have installed Adabas Review 4.5, or if you have loaded a new Adabas Review repository file, you must initialize

- the user profile system for controlling user access to Adabas Review; and
- the Adabas Review data file, which designates a DBID and SVC for the Adabas Review repository, and installs the supplied Adabas Review reports.

Initializing the Adabas Review User Profile System

► To initialize the Adabas Review user profile system

- 1 Access Natural as you normally do.
- 2 At the NEXT prompt, log on to the library SYSREVDDB.


 **Note:** Wherever the NEXT prompt is specified, the command can also be issued from the Natural main menu command line.

- 3 At the NEXT prompt, enter `INSTALL UP` and press `ENTER`.

This initializes the Adabas Review user profile system and adds one default user record to the system. You must have this initial user defined in order to enter Adabas Review.

The following message appears when the user profile subsystem has been successfully installed:

```
Default user profile installed
```

 **Note:** The Adabas Review default user profile is delivered with the HUB / DBID set to zero. When running Adabas Review in local mode, this setting causes the initial HUB / DBID to be set to the same value as the LFILE. However, when starting Adabas Review in hub mode, you need to modify the default user profile and reset the zero to the `ADARUN REVIEW=hubid value`. You can also manually override the default user profile HUB / DBID value by issuing `HUB=hubid` from the command line after you log on to Adabas Review.

- 4 At the NEXT prompt, enter `MENU` and press `ENTER` to display the main menu:

```

20:35:09          A D A B A S  -  R E V I E W          2009-04-30
                                Main Menu              HUB=15690

      Code          Description
      ----          -
      AA           Available Adabas Nuclei
      AO           Adabas Online System
      EB           Edit Buffer Pool Report
      EL           Edit Pulse Report
      ER           Edit Report Definition
      ET           Edit Target Definition
      EX           Edit Cluster Services Report
      LH           List History Reports
      LR           List Report Definitions
      LS           List Started Reports
      LT           List Target Definitions
      UP           User Profiles
      ----          -

Command: _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Fin                                Fin  ←
←

```

Initializing the Adabas Review Data File

 **Note:** If you are using a repository from version 4.4, omit this step.

▶ To initialize the Adabas Review data file

- 1 At the main menu command line, enter `INSTALL DB` and press `ENTER` to display the Initialization Process window:

```
09:34:30          ***** R E V I E W *****          2009-03-10
                          Main Menu

+-----+
|               Initialization Process               |
|                                                     |
| You are about to save information into the         |
| REVIEW repository.                               |
|                                                     |
| The DBID/FNR of the REVIEW repository is          |
| currently set to: DBID: 221                       |
|                      FNR: 12                      |
|                                                     |
| If this is correct enter 'YES' to continue,      |
| else press enter to cancel: ____                |
|                                                     |
+-----+

REVO0001 - WELCOME TO REVIEW RUNNING UNDER COMPLETE
Command: install db_
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9--PF10--PF11--PF12---
      Help      Exit                                  Fin
```

2 Confirm or cancel the initialization.

You are prompted to confirm the DBID and FNR of the Adabas Review repository. If the DBID or FNR are not correct, you may cancel the initialization.

- To confirm the initialization, enter YES at the prompt.
- To cancel the initialization, press ENTER.

If the initialization is confirmed, the Default Target Definition screen appears:

```

14:36:22          ***** R E V I E W *****          2009-03-10
                    Main Menu

+-----+
+-----+
|               Default Target Definition               |
|                                                       |
| Please enter the appropriate SVC and Version         |
| for Adabas DBID 222:                                |
|                                                       |
|           Adabas SVC ..... ____                    |
|           Adabas Version .. ____                    |
|                                                       |
+-----+
+-----+

Command: install db
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
          Help           Exit                                     Fin

```

You are prompted to enter the Adabas SVC number and the Adabas version number of the database designated as the Adabas Review repository.

- 3 Type in the SVC and version number, and press ENTER.

Based on the information you provide, Adabas Review creates the default target definition and displays a message.

The Initialization Process screen appears as shown in the following example:

```

14:37:46          ***** R E V I E W *****          2009-03-10
                    Initialization Process
REV00104 - NOW CREATING DEFAULT TARGET DEFINITION
REV00054 - NOW CREATING SAMPLE REPORT SUMMARY REPORT BY FILE
REV00054 - NOW CREATING SAMPLE REPORT EXCEPTIONAL RESPONSE CODES
REV00054 - NOW CREATING SAMPLE REPORT LONG RUNNING COMMANDS
REV00054 - NOW CREATING SAMPLE REPORT COMMANDS BY HOUR
REV00054 - NOW CREATING SAMPLE REPORT RATE OF COMMANDS AND IOS BY HOUR
REV00054 - NOW CREATING SAMPLE REPORT RATE OF COMMANDS AND IOS BY DATE
REV00054 - NOW CREATING SAMPLE REPORT NATURAL SUMMARY
REV00054 - NOW CREATING SAMPLE REPORT WHO IS USING NATURAL
REV00054 - NOW CREATING SAMPLE REPORT NATURAL PROGRAM TRACE
REV00054 - NOW CREATING SAMPLE REPORT WHO USES SYSMAIN
REV00054 - NOW CREATING SAMPLE REPORT TRANSACTION COUNT BY JOB
REV00054 - NOW CREATING SAMPLE REPORT TRANSACTION COUNT BY JOB-NATAPPL
REV00054 - NOW CREATING SAMPLE REPORT TRANSACTION COUNT BY JOB-USER
REV00054 - NOW CREATING SAMPLE REPORT TRANSACTION COUNT BY NATURAL

REV00006 - PRESS 'ENTER' TO CONTINUE INITIALIZATION PROCESS
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---

```

The Adabas Review supplied reports are automatically added to the Adabas Review repository. As each report is added, a line is added to the Initialization Process screen. Each time the screen fills, you receive the following prompt:

```
Press ENTER to continue the initialization process
```

- 4 Press ENTER as requested; continue until all Adabas Review supplied reports are initialized.

After all the reports are initialized, you are returned to the Adabas Review main menu.

Accessing Adabas Review

▶ To access Adabas Review

- 1 Access Natural as you normally do.
- 2 At the NEXT prompt, log on to the library SYSREVDB (type LOGON SYSREVDB).



Note: Wherever the NEXT prompt is specified, the command can also be issued from the Natural main menu command line.

- 3 At the NEXT prompt, type MENU and press ENTER to access the Adabas Review main menu:


```

20:35:09          A D A B A S - R E V I E W          2009-04-30
                   Main Menu                          HUB=15690

      Code          Description
      ----          -
      AA           Available Adabas Nuclei
      AO           Adabas Online System
      EB           Edit Buffer Pool Report
      EL           Edit Pulse Report
      ER           Edit Report Definition
      ET           Edit Target Definition
      EX           Edit Cluster Services Report
      LH           List History Reports
      LR           List Report Definitions
      LS           List Started Reports
      LT           List Target Definitions
      UP           User Profiles
      ----          -

Command: _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Fin                                Fin  ←
←

```

Verifying the Installation

To confirm that Adabas Review is correctly installed, the installation is verified each time you enter the Adabas Review online subsystem.

The following errors may be detected during processing:

Under Com-plete

USER BUFFER NOT LARGE ENOUGH

Explanation The user buffer extension is not large enough for Adabas Review to pass data to Adabas. The parameter LRVINFO was not set correctly in the link routine.

Action Use the procedure in the installation section of the documentation to correctly install the Adabas Review link routine exit in the link routine.

ADAPRM IS MISSING FROM THE PARAMETER LIST

- Explanation** The Natural ADAPRM area was not passed in the parameter list for the Adabas call. The value ADAPRM=ON was not set correctly in the parameter module (NATPARM) for the Natural nucleus currently executing.
- Action** Use the procedure in the installation section of the documentation to correctly add ADAPRM=ON to the Natural NATPARM parameter module.

1) USER BUFFER EXTENSION NOT LARGE ENOUGH AND 2) ADAPRM IS MISSING FROM THE PARAMETER LIST

- Explanation**
- 1) The user buffer extension is not large enough for Adabas Review to pass data to Adabas. The parameter LRVINFO was not set correctly in the Adabas/Com-plete link routine.
 - 2) The Natural ADAPRM area was not passed in the parameter list for the Adabas call. The value ADAPRM=ON was not set correctly in the parameter module (NATPARM) for the Natural nucleus currently executing.
- Action**
- 1) Use the procedure in the installation section of the documentation to correctly install the Adabas Review link routine exit in the Adabas/Com-plete link routine.
 - 2) Use the procedure in the installation section of the documentation to correctly add ADAPRM=ON to the Natural NATPARM parameter module.

THE Adabas REVIEW LINK ROUTINE IS NOT CORRECTLY INSTALLED

- Explanation** The Adabas Review link routine exit is not installed in the copy of the link routine currently being executed.
- Action** Use the procedure in the installation section of the documentation to correctly install the Adabas Review link routine exit in the link routine.

THE Adabas REVIEW REPOSITORY HAS NOT BEEN INITIALIZED

- Explanation** Adabas Review could not initialize because the repository file has not been initialized.
- Action** Use the procedure described in section *Initializing the Adabas Review Data File* to correctly initialize the repository file.

Under CICS

USER BUFFER NOT LARGE ENOUGH

- Explanation** The user buffer extension is not large enough for Adabas Review to pass data to Adabas. The parameter LRVINFO was not set correctly in the Adabas/CICS link routine.
- Action** Use the procedure in the installation section of the documentation to correctly install the Adabas Review link routine exit in the Adabas/CICS link routine.

ADAPRM IS MISSING FROM THE PARAMETER LIST

Explanation The Natural ADAPRM area was not passed in the parameter list for the Adabas call. The value ADAPRM=ON was not set correctly in the parameter module (NATPARM) for the Natural nucleus currently executing.

Action Use the procedure in the installation section of the documentation to correctly add ADAPRM=ON to the Natural NATPARM parameter module.

1) USER BUFFER EXTENSION NOT LARGE ENOUGH AND**2) ADAPRM IS MISSING FROM THE PARAMETER LIST**

Explanation 1) The user buffer extension is not large enough for Adabas Review to pass data to Adabas. The parameter LRVINFO was not set correctly in the Adabas/CICS link routine.

2) The Natural ADAPRM area was not passed in the parameter list for the Adabas call. The value ADAPRM=ON was not set correctly in the parameter module (NATPARM) for the Natural nucleus currently executing.

Action 1) Use the procedure in the installation section of the documentation to correctly install the Adabas Review link routine exit in the Adabas CICS link routine.

2) Use the procedure in the installation section of the documentation to correctly add ADAPRM=ON to the Natural NATPARM parameter module.

THE Adabas REVIEW LINK ROUTINE IS NOT CORRECTLY INSTALLED

Explanation The Adabas Review link routine exit is not installed in the copy of the Adabas/CICS link routine currently being executed.

Action Use the procedure in the section *Installation (Phase 2)* to correctly install the Adabas Review link routine exit in the Adabas/CICS link routine.

THE Adabas REVIEW REPOSITORY HAS NOT BEEN INITIALIZED

Explanation Adabas Review could not initialize because the repository file has not been initialized.

Action Use the procedure described in section *Initializing the Adabas Review Data File* to correctly initialize the repository file.

6 Operations

- Assigning Logical Units 40
- Processing ABENDs 41
- Adding Adabas Startup Statements at Installation 42
- Files Used by Adabas Review 43
- Editing the RVUEXI Parameter File 47
- Command Logging Considerations 48
- Summary Logging Considerations 49
- Modifying Configuration Parameters 51
- User Exit 5 (Adabas Review Hub Event Handler) 53
- Adabas Review Natural User Exits 55

This section of the documentation describes operational procedures and processes for Adabas Review after it has been installed and initialized.

Assigning Logical Units

Adabas Review uses the following logical units (SYS numbers):

SYSnnn	File Name	Explanation
SYS005	RVUPARM	Parameter file (primarily used in batch mode)
SYS006	RVUEXI	Parameter file
SYS006	RVUSEQ	Command log input (batch mode only)
SYS007	RVUAUT1	Autostart parameter file 1
SYS007	RVUAUT2	Autostart parameter file 2
SYS007	RVUFLD	User-defined fields
SYS008	RVUALT	Alternate history sequential data set
SYS019	RVUEXP	Printer for RVUEXI parameters
SYS020	RVUPRT0	Printer 0 (Adabas Review statistics)
SYS021	RVUPRT1	Printer 1 (Adabas Review reports)
SYS022-SYS029	RVUPRTn	Printers 2-9 (Adabas Review reports)
SYS031	RVUCOP1	Copy file 1 output
SYS032-SYS039	RVUCOPn	Copy files 2-9 output
SYS041	RVUCARD	GENCARD output

Any of these logical units may be changed if they conflict with logical units already in use.

▶ To change logical unit numbers used by Adabas Review

- 1 Edit the supplied source member RAOSLUBS.A in the Adabas Review library.
This member is self-documenting.
- 2 Make the necessary changes.
- 3 Adapt the LNKLUBS.X JCS and use it to assemble and link RAOSLUBS.A.

Processing ABENDs



Note: This section applies only to Adabas Review in local mode.

If Adabas Review processing terminates abnormally (ABENDs) in RAOSLOCL, the Adabas Review STXIT routine is given control.

This routine traps the ABEND, and prints diagnostic information and a dump to expedite the analysis and resolution of the ABEND to ADASNAP.

The routine also disables Adabas Review processing for the remainder of the Adabas nucleus session.

If you need help resolving the ABEND, contact your Software AG technical support representative with the printed information.

ABEND Protection for Adabas

Adabas is protected from termination if Adabas Review ABENDs. Adabas processing will continue without interruption.



Note: Although Adabas will be temporarily unavailable during dump processing, it will continue processing as usual once the dump has completed.

Message to Adabas Review Users

Users attempting to access Adabas Review following a trapped ABEND will receive the following message:

```
Adabas Review not installed on database
```

Messages to the Console

The progress of the error handling routine is reported by messages written to the console. The example sequences of console messages provided in this section are sent during ABEND processing:

```
REVESTAE - REVIEW ESTAE EXIT DRIVEN
REVESTAE - REVIEW NOW DISABLED
REVESTAE - ABEND 000C1000 PSW 078D2000 80129E98
REVESTAE - R0 00000002 - R1 0D652DD0 - R2 000FD240 - R3 000FBCC0
REVESTAE - R4 00129C48 - R5 0D50AFA8 - R6 0D6E8000 - R7 001331F8
REVESTAE - R8 0D50B0E8 - R9 800E93E0 - R10 00042000 - R11 0D5007E0
REVESTAE - R12 80128C48 - R13 00128D68 - R14 001294BA - R15 8000DD10
REVESTAE - DUMP HAS BEEN TAKEN
```

REV20126 - REVIEW SUB-TASK DETACHED
 REV20129 - HISTORY SUB-TASK DETACHED

Adding Adabas Startup Statements at Installation



Note: This section applies only to Adabas Review in local mode.

Statements must be added to the Adabas startup job to accommodate Adabas Review. These statements are added during the installation of Adabas Review.

The added statements control many of the operating features of Adabas Review. Some identify parameter files that may be edited by the Adabas Review administrator.

The startup statements may be edited, or additional statements may be needed, depending upon the needs of your site. However, deleting any of these statements will affect the functioning of Adabas Review and is, therefore, not recommended.

Adabas Startup Statements

For z/VSE, the following statements are added to the Adabas startup job stream during Adabas Review installation:



Note: The logical units shown in the example below may be reassigned if there are conflicts with your site-specific logical units. Refer to the section [Assigning Logical Units](#) for more information.

```
// ASSGN SYS005,IGN          RVUPARM - batch parameters
// ASSGN SYS007,DISK,VOL=vvvvvv,SHR  RVUAUT1, RVUAUT2
// ASSGN SYS008,DISK,VOL=vvvvvv,SHR  RVUALT - alt history file
// ASSGN SYS020,SYSLST        RVUPRT0 - printer
// ASSGN SYS021,SYSLST        RVUPRT1 - printer
// ASSGN SYS022,SYSLST        RVUPRT2 - printer
// ASSGN SYS023,SYSLST        RVUPRT3 - printer
// ASSGN SYS041,DISK,VOL=vvvvvv,SHR  RVUCARD - GENCARD
// ASSGN SYS006, DISK,VOL=vvvvvv,SHR  RVUEXI - operating env parameters
// ASSGN SYS019,SYSLST        RVUEXP - EXI parameters printer
*
// DLBL RVUAUT1,'REVvrs.AUT01',0
// EXTENT SYS007,vvvvvv,1,0,nnnn,5
// DLBL RVUAUT2,'REVvrs.AUT02',0
// EXTENT SYS007,vvvvvv,1,0,nnnn,5
*
// DLBL RVUFLD,'REVvrs.USER.FIELDS',0
// EXTENT SYS007,vvvvvv
*
// DLBL RVUCARD,'REVvrs.CARD.FILE',0
```



```
// EXTENT SYS041,vvvvvv,1,0,nnnn,5
*
// DLBL RVUALT,'REVvrs.ALT.FILE',0
// EXTENT SYS008,vvvvvv,1,0,nnnn,30
*
// DLBL RVUEXI,'REVvrs.EXI.PARAMETERS',0
// EXTENT SYS006,vvvvvv,1,0,nnnn,5
```

where *vvvvvv* is the volume serial number of the target disk, *nnnn* is the relative track number, and *vrs* is the version, revision, and system maintenance level of Adabas Review.

The Adabas startup statements listed above identify files that are used by Adabas Review. Each of these files is described in the following section *Files Used by Adabas Review*.

Files Used by Adabas Review

RVLOG01 and RVLOG02 Command Logging Files

RVLOG01 and RVLOG02 are the default names for the sequential command logging files. Each report performing command logging must reference a unique file name prefix and a number of command log files associated with that file name prefix.



Note: All command log data sets for a particular report must be the same size.

Adabas Review allows each report to have up to 99 command log files and writes to these files in sequential order. DLBL, EXTENT, and ASSGN statements must be added to the JCS for each command log file. The names of these command log files are made up of the file name prefix and a sequential number. The data sets for these command log files are created when they are opened.

Refer to the section *Command Logging Considerations*, elsewhere in this guide, for more information.

RVSUM01 and RVSUM02 Summary Logging Files

RVSUM01 and RVSUM02 are the default names for the sequential summary logging files. Each report performing summary logging must reference a unique file name prefix and a number of summary log files associated with that file name prefix.



Note: All summary log data sets for a particular report must be the same size.

Adabas Review allows each report to have up to 99 summary log files and writes to these files in sequential order. DLBL, EXTENT, and ASSGN statements must be added to the JCS for each summary log file. The names of these summary log files are made up of the file name prefix and a sequential number. The data sets for these command log files are created when they are opened.

Refer to the section *Summary Logging Considerations*, elsewhere in this guide for more information.

RVUALT History File

Adabas Review reports may specify whether the data accumulated by the report will also be written to the Adabas Review repository. Historical data is useful for monitoring database performance and for performing trend analysis.

The parameters that determine whether Adabas Review writes historical data are set when a user creates or edits a report definition. These history parameters appear on the Report Options screen of the Edit Report (ER) function.

If historical data is to be written by a report running in batch mode, the history parameters make up the COPY statement.

A RVUALT job control statement can identify an alternate sequential file to which historical data may be written when it cannot be written to the Adabas Review repository.

- In hub mode, the Adabas Review hub startup JCS contains a RVUALT statement.
- In local mode, the Adabas startup JCS must be modified to include a RVUALT statement during the Adabas Review installation procedure.

RVUALT data sets must be allocated:

- In hub mode, a separate RVUALT data set must be allocated for each Adabas Review hub.
- In local mode, if Adabas Review is installed on multiple databases, an RVUALT data set must be allocated for each database.

The RVUALT history file is allocated using the job DBFILES.

Adabas Review receives a response code 148 (Adabas not active) and writes the data to the file specified by the RVUALT job control statement (if it has been assigned in the job stream) in situations where the Adabas Review repository is

- unavailable (in hub mode).
- on the same database that is being monitored (in local mode). The response code is returned when the database is brought down and Adabas Review tries to write the historical data.

The next time the Adabas Review hub is started (in local mode, the next time Adabas is started), another subtask is started to copy the historical data from the RVUALT file to the Adabas Review repository.

RVUAUT1 and RVUAUT2 Report Definition Data Sets

RVUAUT1 and RVUAUT2 are data sets that contain the report definition control statements for autostarted reports. Adabas Review generates the statements and writes them to these files. When Adabas is initialized, the reports are started automatically.

Under z/VSE, this file is created by the installation job DBFILES.X.



Note: In a future release of Adabas Review, RVUAUT2 will no longer be supported. RVUAUT1 only will be used for autostarted reports.

RVUCARD Data Set for the GENCARD Command

RVUCARD is a data set used by the GENCARD command. The GENCARD command creates batch parameter statements from report definitions created online.

The command requires the user to supply DLBL, EXTENT, and ASSGN statements, and the generated statements are written to the corresponding file.

RVUEXI Parameter File

RVUEXI is a parameter file that contains parameters to control the Adabas Review operating environment. The Adabas Review administrator may edit the RVUEXI parameters according to the specific needs of the site.

Refer to the section [Editing the RVUEXI Parameter File](#) for more information.

RVUEXP Companion Output File

RVUEXP is a companion file to RVUEXI and if specified, any parameter processing errors encountered in RVUEXI will be written to the RVUEXP output file.

RVUFLD User Field Parameter Data Set

The RVUFLD data set contains parameter control statements for creating user-defined fields. Parameters in this data set define the length, type, and location of reporting fields to be determined by the user.

RVUPARM Dummy Data Set

The recommended procedure is to set RVUPARM as a dummy data set by assigning SYS005 to IGN. In previous releases, batch parameter statements were read from this file. Because these statements may now be generated using the GENCARD command, you no longer need to code batch parameters manually. Parameters may be coded in this data set if desired, and Adabas Review will access this data set prior to accessing data sets specified by RVUAUT1 and RVUAUT2.



Note: When RVUPARM has been "dummied", the message **REV20164 - Open failed for RVUPARM** is displayed. In this case, the message is normal and should be ignored. The message does not occur if instead you create a RVUPARM data set that contains only an asterisk.

RVUPRTnn Logical Printer Files

RVUPRT00 for Adabas Review Statistics

RVUPRT00 is the Adabas Review logical printer for statistics about Adabas Review operations, such as number of reports, number of records processed, etc.

RVUPRTnn Files for Reports

RVUPRT01, 02, . . . nn

RVUPRT01 and above are Adabas Review logical printers used for reports. One logical printer is shared by all summary reports; each detail report requires its own logical printer. A job control statement corresponding to each logical printer must be added to the Adabas Review hub (in local mode, to the Adabas) startup job control (JCS).

Assignment of logical printers to reports depends on the order in which the reports are started:

- If the first report started is a *summary* report, RVUPRT01 is used for all summary reports.
- If the first report is a *detailed* report, RVUPRT01 is assigned to the detailed report, and another logical printer is used for summary reports. When a detail report is purged, the corresponding printer number is freed. The next detail report started will reuse the lowest available printer number.

Editing the RVUEXI Parameter File

The RVUEXI file contains parameters to control the Adabas Review operating environment. The Adabas Review administrator may edit the following RVUEXI parameters according to the specific needs of the site:



Note: Default values are underlined in the following tables.

RVUEXI User-Specified Parameter

Parameter	Possible Values	Default
UIDT-CELLS	100-10000	<u>1000</u>

The user ID table is managed using a hashing algorithm. This value is numeric and specifies the number of 8-byte cells that should be allocated to the user ID table manager.

RVUEXI Timeout Parameters

Parameter	Possible Values	Default
UCMD-TIMEOUT	0-999	<u>60</u>

A small reentrant storage area is allocated for each active user of the Adabas Review online system (LIST, VIEW, START, PURGE reports functions). This area is deallocated when the user finishes each online request.

However, if the user's Natural session terminates abnormally during an Adabas Review operation, the Adabas Review nucleus may not have the opportunity to deallocate the reentrant area.

Specifying the UCMD-TIMEOUT parameter gives the Adabas Review nucleus a timeout value after which these inactive areas are deallocated. The timeout value is numeric and is specified in minutes.

Parameter	Possible Values	Default
UIDT-TIMEOUT	1-999	<u>60</u>

To report on the field TPTRANCT, Adabas Review must maintain a work area for each user that accesses Adabas. This area is called the user ID table.

If this field is specified in a report, the facility is activated and an area is allocated when Adabas Review receives the first call from each user. The area is deallocated when Adabas Review receives an Adabas CLOSE (CL) command for that user.

However, if the user's application does not issue a CL during termination, Adabas Review is unaware that the session has terminated.

The `UIDT-TIMEOUT` parameter is used to expire inactive user ID table elements. If the field `TPTRANCT` is *not* specified in any active reports, Adabas Review will *not* maintain user ID table elements for each user. This value is numeric and specifies the timeout value in minutes.

Command Logging Considerations

This section discusses administrative considerations when performing Adabas Review command logging.

Setting Up Command Logging

Adabas Review provides you with options for determining how command logging is processed for reports. However, the Adabas Review administrator must complete the following tasks to set up the Adabas Review environment so that command logging can take place:

1. Allocate command log data sets. Command log data sets must be allocated for reports.
2. Add job control statements to the Adabas Review hub startup JCS (hub mode) or the Adabas nucleus startup JCS (local mode).

Each report that performs command logging must have a command log file assigned to it. For each command log file, there must be a corresponding job control statement in the startup JCS.

The name must be a five-character name followed by a sequential number (01, 02, etc.) corresponding to the number of command logs. For example, if the name is "CMLOG" and there are two data sets to be defined, two statements are required with names as follows: CMLOG01 and CMLOG02. The five-character name is referenced by the report in the File command logging report option. The total number of data sets is referenced by the report in the Num of Logs command logging report option. For more information about these logging options, read *Logging Options*, in the *Adabas Review User's Guide*.

Refer to the section [RVLOG01 and RVLOG02 Command Logging Files](#), elsewhere in this guide, for more information.

Using the Command Logging User Exit

Adabas Review writes to command log files in sequential order. When a command log file is filled, Adabas Review closes the file, switches to the next sequential file, and continues logging. When all files have been filled, Adabas Review switches back to the oldest file to log data. If a command logging user exit is *not* specified, Adabas Review will write over the log data in the file containing the oldest data.

A command logging user exit can be specified so that the data contained in the command log file can be copied to a new file before the command log file is overwritten with new command log data. This user exit will be called each time a command log file is closed or opened, but it is only called if you reference it in the User Exit report logging option. For more information about this logging option, read *Logging Options*, in the *Adabas Review User's Guide*.

LOGUEXIT Sample User Exit

The source library member LOGUEXIT contains sample code for the user exit that processes command logs. You may modify this exit so that it conforms to your site requirements, and you can include the exit name in the User Exit logging option on the **Report Options** screen of your report definition.

End-of-File Marker Position

When a command log file is opened, the user exit checks the position of the end-of-file marker to determine if there is any data in the command log file.

- If the position indicates that there is no data in the file, Adabas Review writes command log data to the file.
- If the position indicates that there is data in the file, Adabas Review sends a message to the operator asking whether it should wait until the copying of the command log is completed or begin writing to the command log file and overwrite the existing data.

Summary Logging Considerations

This section discusses administrative considerations when performing Adabas Review summary logging.

Setting Up Summary Logging

Adabas Review provides you with options for determining how summary logging is processed for reports. However, the Adabas Review administrator must complete the following tasks to set up the Adabas Review environment so that summary logging can take place:

1. Allocate summary log data sets. Summary log data sets must be allocated for reports.
2. Add job control statements to the Adabas Review hub startup JCS or the Adabas nucleus startup JCS (local mode).

Each report that performs summary logging must have a summary log file assigned to it. For each summary log file, there must be a corresponding job control statement in the startup JCS.

The name must be a five-character name followed by a sequential number (01, 02, etc.) corresponding to the number of summary logs. For example, if the name is "SMLOG" and there are two data sets to be defined, two statements are required with names as follows: SMLOG01 and SMLOG02. The five-character name is referenced by the report in the Summary File summary logging report option. The total number of data sets is referenced by the report in the Num of Logs summary logging report option. For more information about these logging options, read *Logging Options*, in the *Adabas Review User's Guide*.

Refer to the section [RVSUM01 and RVSUM02 Summary Logging Files](#), elsewhere in this guide, for more information.

Using the Summary Logging User Exit

Adabas Review writes to summary log files in sequential order. When a summary log file is filled, Adabas Review closes the file, switches to the next sequential file, and continues logging. When all files have been filled, Adabas Review switches back to the oldest file to log data. If a summary logging user exit is *not* specified, Adabas Review will write over the log data in the file containing the oldest data.

A summary logging user exit can be specified so that the data contained in the summary log file can be copied to a new file before the summary log file is overwritten with new summary log data. This user exit will be called each time a summary log file is closed or opened, but it is only called if you reference it in the Log Full Exit report logging option. For more information about this logging option, read *Logging Options*, in the *Adabas Review User's Guide*.

LOGUEXIT Sample User Exit

The source library member LOGUEXIT contains sample code for the user exit that processes summary logs. You may modify this exit so that it conforms to your site requirements, and you can include the exit name in the Log Full Exit logging option on the **Report Options** screen of your report definition.

REVCLCOP Sample Copy Job

When a summary log file is closed, the user exit starts the summary log copy job. The JCS library member REVCLCOP.X contains a sample summary log copy job. This job copies the contents of a filled summary log file to another device and appends the new data to existing data. This job also reinitializes the end-of-file marker in the summary log file.

End-of-File Marker Position

When a summary log file is opened, the user exit checks the position of the end-of-file marker to determine if there is any data in the summary log file.

- If the position indicates that there is no data in the file, Adabas Review writes summary log data to the file.
- If the position indicates that there is data in the file, Adabas Review sends a message to the operator asking whether it should wait until the copying of the summary log is completed or begin writing to the summary log file and overwrite the existing data.

Modifying Configuration Parameters

The Adabas Review administrator can modify configuration parameter values in the Natural text member CONFIGDB.

▶ To access and modify these parameters

- 1 At the Natural NEXT prompt, type `LOGON SYSREVDDB` and press `ENTER`.
- 2 Type the command `EDIT CONFIGDB` and press `ENTER`. Modify the parameters as required.
- 3 Type `SAVE` and press `ENTER` to save the changes.
- 4 Type `MENU` at the prompt to return to Adabas Review.

CONFIGDB File Parameter Description

CONFIGDB, which contains Adabas Review parameters, is saved in the Natural library SYSREVDDB.

Parameter	Possible Values	Default
CURSOR-POSITION	<u>B</u> OT <u>T</u> OP	<u>B</u> OT

Specifies whether the cursor is placed on the command line (BOT) in list displays, or on the SEL field (TOP).

Parameter	Possible Values	Default
DECIMAL-CHAR	NAPARM DC= <i>value</i>	.

Specifies the decimal character to use when generating Adabas Review reports. The value specified should match the value specified for the NATPARM DC parameter. To determine the current setting of the NATPARM DC parameter, issue GLOBALS at the NEXT prompt. The Adabas Review default value for DECIMAL-CHAR is a period ('.').

Parameter	Possible Values	Default
PC-FILE	'text'	'DOWNLOAD-PC-FILE-5'

Specifies the value to be used in the DOWNLOAD statement in the Adabas Review-generated programs. The value specified must be delimited with single apostrophes. The field is alphanumeric, maximum 20 characters.

Parameter	Possible Values	Default
RVXB-MESSAGE	<u>Y</u> ES <u>N</u> O	<u>Y</u> ES

Specifies whether to display error messages about the incorrect installation of the Adabas Review link routine exits during installation verification.

Parameter	Possible Values	Default
UBAR	any valid character	

Specifies the character to be used in maps as the vertical border. Any character recognized by your system is valid; the default value is "|".

Parameter	Possible Values	Default
CLOSE-DBID	<u>Y</u> ES <u>N</u> O	<u>N</u> O

Specifies whether to issue a close (CL) command to the old Adabas Review database when a new database is accessed with the HUB= (DBID=) command.

Parameter	Possible Values	Default
REVIEWDB-UEX	<i>name</i>	exit not enabled

Specifies the name of the site-dependent Natural routine to be called for validation of a user's access to an Adabas Review function.

Refer to Natural source member N-USEXIT for more information on the calling and processing conventions for this exit.

Parameter	Possible Values	Default
MAXIMUM-MAXK	0 nnnn	0

Specifies the maximum value that can be specified for the report option, Max K. The Max K value determines the maximum amount of storage available for a specific report.

A value of 0 (the default) indicates that the Max K option is not restricted.

When specifying a value, MAXIMUM-MAXK must be 4 or greater for z/VSE.

Parameter	Possible Values	Default
OPEN-DBID	<u>Y</u> ES <u>N</u> O	<u>N</u> O

Specifies whether an open (OP) command is issued to the new Adabas Review database when a new database is accessed with the HUB= (DBID=) command.

User Exit 5 (Adabas Review Hub Event Handler)

User exit 5 is called by the Adabas nucleus when an *event* occurs with the Adabas Review hub. User exit 5 must be specified in ADARUN parameter UEX5 in the Adabas nucleus startup job. An event is defined as:

- a connection made with the Adabas Review hub during Adabas session open;
- a connection ended with the Adabas Review hub during Adabas session close; or
- a non-zero return code received from the send operation for a command log record. When buffering is active, this return code is provided once for a whole buffer and it is possible that only parts of the buffer were not transferred correctly.

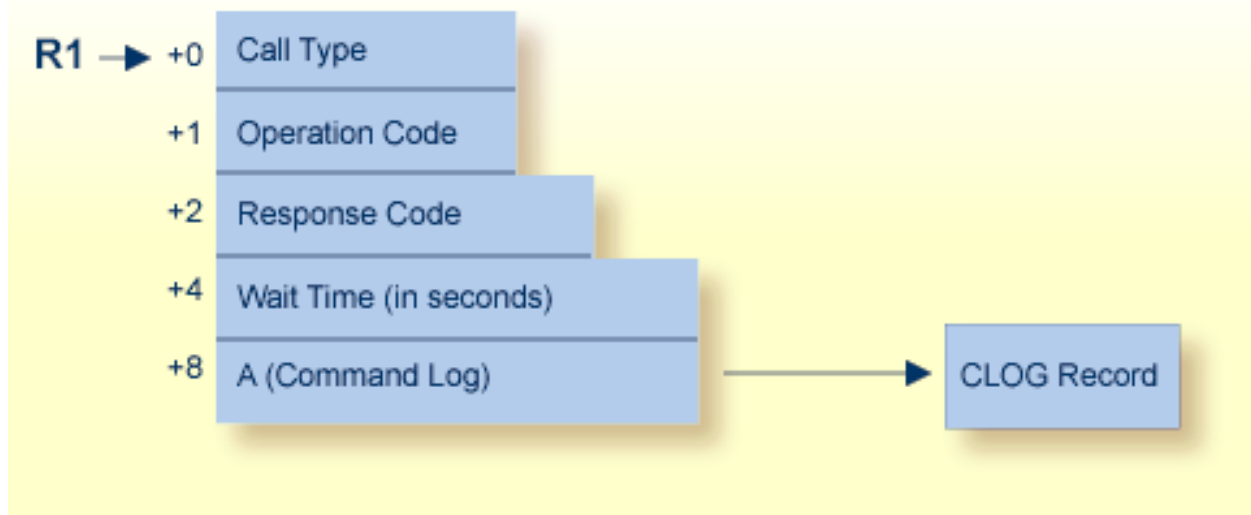
The exit is invoked with AMODE=31 and should return control in the same state.

The exit is required to process logging errors. It determines how the failure is handled. The parts of the buffer that were not logged and the response code received from the Adabas Review hub logging request are provided to assist in making the determination.

- Input Parameters
- Output Parameters

Input Parameters

On entry, register 1 points to the following parameter list:



Parameter	Usage
0(R1)	Exit call indication. The value of this byte can be: <ul style="list-style-type: none"> ■ "O" -- connection with Adabas Review hub opened; ■ "C" -- connection with Adabas Review hub closed; or ■ "L" -- sending logging error to Adabas Review hub.
1(R1)	Action to handle a logging error (ignored for open and close). The exit must provide one of the following values for this field in the parameter list for a logging error: <ul style="list-style-type: none"> ■ "W" -- wait for a specified time and then retry; ■ "R" -- retry logging operation immediately; or ■ "I" -- ignore the logging failure and continue without consequence.

Parameter	Usage
2(R1)	Response code for logging errors. This response code is the same as the Adabas response code in the <i>Adabas Messages and Codes</i> documentation.
4(R1)	Fullword where the exit must provide a wait time (in seconds) for the logging failures that are to be retried after waiting.
8(R1)	Address of the command log record that the Adabas nucleus was attempting to send to the Adabas Review hub.

Other Register Values at Entry

Value	Description
R13	Save area of calling Adabas nucleus routine.
R14	Return address in Adabas nucleus.
R15	Entry point address for exit.

Output Parameters

- For logging errors, the exit is required to set a value in the *operation* field. If the wait value (W) is chosen, the exit is also required to provide a non-zero time value.
- Register 15 should be set to zero. All other registers should be returned intact.

Adabas Review Natural User Exits

Adabas Review has two Natural user exits. These exits are found in the Adabas Review system library in Natural, and may be modified by using the Natural editor. They are applicable to both the Adabas Review and Adabas Review Data Communication systems.

P-UEXIT1

Use:	You may place coding in this program to allow for site-specific needs.
Invoked:	This program is invoked when the online portion of Adabas Review is entered.
Example 1:	Setting colors on (SET CONTROL 'T3279').
Example 2:	Turning the PC mode on or off.
Remark:	This program must <i>not</i> alter the Natural stack, and it must end with a STOP command.

P-UEXIT2

Use:	You may place coding in this program to alter the processing that occurs when terminating Adabas Review.
Invoked:	This program is invoked when the online portion of Adabas Review is terminated.
Example 1:	Returning to Natural rather than terminating your session.
Example 2:	Logging on to another Natural application.
Example 3:	Returning to a previous Natural application (using <code>SETUP/RETURN</code>).

7

ADARUN Parameters for Adabas Review in z/VSE

▪ ADARUN Parameter Syntax	58
▪ PROGRAM : Program to Run	59
▪ REVIEW : Adabas Review Control	60
▪ CT : Command Timeout Limit	61
▪ FORCE : Allow Nucleus Database ID or Review Hub Table Entry Overwrite	62
▪ LOCAL : Local Nucleus or Adabas Review Hub	63
▪ NAB : Number of Attached Buffers	64
▪ NC : Number of Command Queue Elements	65
▪ REVFILTER : Review Record Filtering Control	66
▪ REVLOGBMAX: Logged Buffer Size Limit for Review	66
▪ REVLOGMAX: Total Logged Buffer Size Limit for a Review Command	67
▪ SVC : SVC Number	67

ADARUN performs the following functions:

- Loads the ADAIOR module, which performs all database I/O and other operating-system-dependent functions.
- Interprets the ADARUN parameter statements; then loads and modifies the appropriate Adabas nucleus or utility modules according to the ADARUN parameter settings.
- Transfers control to Adabas.

The ADARUN statement, normally a series of entries each specifying one or more ADARUN parameter settings, is specified in the VSE CARD data set. For more specific job information, refer to the appropriate section of this documentation.

The ADARUN control statement defines and starts the Adabas operating environment. The ADARUN control statement also starts Adabas utilities.

The ADARUN parameters described in this section of the documentation apply to Adabas Review in z/VSE environments.

ADARUN Parameter Syntax

The syntax for the ADARUN statement and parameters is:

```
ADARUN parameter=value,...
```

where *parameter=value* is one or more of the ADARUN parameters described in this section.

Any number of blanks is permitted between "ADARUN" and the first parameter, but no blanks are permitted within the *parameter=value* string. Commas (,) must be used as separators. A blank following a *parameter=value* entry indicates the end of the statement.

The literal "ADARUN" must be entered in positions 1-6 of each ADARUN statement. All *parameter=value* entries must end before position 73. Any *parameter=value* entries that would extend beyond position 72 must be coded on a new statement as shown below. The comma following the last *parameter=value* entry of a statement is optional, and is not interpreted as a continuation character. Positions 73-80 are ignored. An asterisk (*) in position 1 indicates a user comment line.

The following example summarizes the ADARUN statement format. The first statement cannot continue beyond position 72. The second statement represents a continuation of the first statement. All ADARUN continuation statements have the same format and restrictions as the first statement.

Positions 1-6	Positions 8-72
"ADARUN"	<i>parameter=value,parameter=value,...</i>
"ADARUN"	<i>parameter=value,...</i>

The ADARUN parameters are summarized in the following table, and described in detail in the section following the table.

Unless noted otherwise, each parameter has a default value that ADARUN uses if the parameter is not specified.

Some parameter names can be abbreviated. The minimum acceptable abbreviation for a given parameter is shown in the following table by the underlined part of the parameter name. Parameter names shown without underlining cannot be abbreviated.

PROGRAM : Program to Run

Parameter	Specify:	Possible Values	Default
<u>PROGRAM</u>	the program to be executed.	see table below	USER

This parameter specifies what to execute. The possible values are described in the following table:

Specify:	To start:
ADACOM	an ADACOM task (used in Adabas Cluster Services and Adabas Parallel Services environments) For more information, refer to your Adabas Cluster Services and Adabas Parallel Services documentation.
ADANUC	an Adabas nucleus For more information about executing an Adabas nucleus, read <i>Adabas Session Execution</i> , in <i>Adabas Operations</i> documentation.
ADAREV	an Adabas Review hub. Specify this in conjunction with the ADARUN REVIEW parameter. For more information, refer to your Adabas Review documentation.
NETWRK	an Entire Net-Work node For more information, refer to your Entire Net-Work documentation.
RENTUSER	a user program to be run using a reentrant Adabas batch/TSO link routine. For more information, refer to your Adabas TP monitor installation documentation.

Specify:	To start:
USER	a user program to be run using a non-reentrant Adabas batch/TSO link routine. For more information, read <i>Linking Applications to Adabas</i> , in <i>Adabas Operations</i> documentation.
<i>utility-name</i>	an Adabas utility Specify an Adabas utility for <i>utility-name</i> . For more information, refer to <i>Adabas Utilities</i> documentation.

Examples

The following example specifies that an Adabas nucleus is running.

```
ADARUN PROGRAM=ADANUC
```

The following example specifies that an Adabas Review hub is running.

```
ADARUN PROGRAM=ADAREV, REVIEW=202
```

The following example specifies that an Entire Net-Work node is running.

```
ADARUN PROGRAM=NETWRK
```

REVIEW : Adabas Review Control

Parameter	Specify . . .	Possible Values	Default
REVIEW	whether to run Adabas Review in local or hub mode specifying the hub ID, or not at all.	NO LOCAL dbid	NO



Note: The parameter name REVIEWHUBID is a synonym for REVIEW, provided to ensure downward compatibility with past Adabas releases. We recommend that you use the parameter name REVIEW instead, wherever possible.

REVIEW controls the use of the Adabas Review product:

Value	Meaning
NO	The default setting. Adabas Review is not started.
LOCAL	Adabas Review is started in local mode running as an extension to ADALOG. This is the only active Adabas Review setting available under z/VM. In local mode, Adabas Review job control statements should be added to the Adabas nucleus startup JCL.

Value	Meaning
	<p>Note: If an Adabas Review load library is not included in the startup JCL, the REVIEW parameter is automatically changed from LOCAL to NO.</p>
dbid	<p>Adabas Review is started in hub mode. The physical database ID that you specify for the hub identifies</p> <ul style="list-style-type: none"> ■ the hub (server) itself (with PROGRAM=ADAREV) that is being started; or ■ from an Adabas nucleus (client), the hub that is the target for Adabas Review processing for that nucleus (with PROGRAM=ADANUC). <p>In hub mode, Adabas Review job control statements should be added to the Adabas Review hub startup JCL.</p>

Dynamic Modification

The setting of the ADARUN PROG=ADANUC,REVIEW=dbid parameter can be changed dynamically using the REVIEWHUBID command from the operator console, the ADADBS OPERCOM REVIEWHUBID function, or the Modify Parameter function of Adabas Online System.

Examples

The following example starts hub 202 for the Adabas Review hub (server) installation.

```
ADARUN PROGRAM=ADAREV,REVIEW=202
```

The following example starts the Adabas nucleus that will log to Adabas Review hub 202 for the Adabas Review (client) installation.

```
ADARUN PROGRAM=ADANUC,REVIEW=202
```

CT : Command Timeout Limit

Parameter	Specify . . .	Minimum	Maximum	Default
CT	the maximum time (seconds) for interregion communication of results from Adabas to the user.	1	16777215	60

For Adabas Review, this is the maximum number of seconds (more precisely, units of 1.048576 seconds) that can elapse from the time an Adabas Review hub command has been completed until the results are returned to the user through the interregion communication (operating-system-dependent).

This parameter is used to prevent a command queue element and attached buffer from being held for a long period of time for a user who has terminated abnormally.

Possible causes of a command timeout are

- user region is swapped out or cannot be dispatched;
- user is cancelled;
- user has low priority in high activity system.

If the CT limit is exceeded,

- the command queue element and attached buffer are released;
- a message ADAM93 is printed; and
- if the user has not terminated, response code 254 is returned to the user program.

Example

The following example permits about 30 seconds to obtain a result through interregion communication from the Adabas Review hub

```
ADARUN PROG=ADAREV,CT=30
```

FORCE : Allow Nucleus Database ID or Review Hub Table Entry Overwrite

Parameter	Specify . . .	Possible Values	Default
FORCE	whether the nucleus or Adabas Review hub can overwrite an existing ID table entry.	YES NO	NO

If running Adabas Review, this indicates whether the Adabas Review hub can overwrite an existing ID table entry. When a Review hub starts up, ADARUN scans the ID table to ensure that no entry exists for the Review hub. You can use the FORCE parameter to indicate whether the Review hub can overwrite an existing ID table entry.

The ID table entry is derived from the database ID and the job name. For Adabas Review, the ID table entry is derived from the Review hub ID (REVIEW=). The ID table entry is deleted when the nucleus terminates normally.

The FORCE parameter allows the nucleus or Adabas Review hub to overwrite the existing ID table entry and access the database.



Caution: Do not use the FORCE parameter unless absolutely necessary, or the integrity of the database could be lost. Ensure that no nucleus or Review hub is active for the ID table entry being overwritten.

Value Meaning

- YES** The nucleus or Adabas Review hub that is starting can overwrite an existing ID table entry. FORCE=YES is required when restarting a session that terminated abnormally with an ADAM98 message. In this case, the ID table still contains an active entry for the nucleus or Review hub. Overwriting the existing entry by specifying FORCE=YES prevents further communication to the overwritten nucleus or hub and causes loss of cross-memory environment resources, which cannot be restored until the next IPL.
- NO** (default) If the ID table contains an entry for the nucleus or Adabas Review hub that is starting, the nucleus is denied access to the database or the Review hub is not permitted to start.



Note: In an Adabas Cluster Services or Adabas Parallel Services environment, the FORCE parameter applies to the NUCID, rather than the database ID, because a cluster nucleus builds an ID table entry for the NUCID.

Examples

The following example specifies that if the ID table contains an active entry for DBID 7, overwrite the entry.

```
ADARUN PROG=ADANUC, FORCE=YES, DBID=7
```

The following example specifies that if the ID table contains an active entry for the Adabas Review hub, overwrite the entry.

```
ADARUN PROG=ADAREV, FORCE=YES, REVIEW=202
```

LOCAL : Local Nucleus or Adabas Review Hub

Parameter	Specify . . .	Possible Values	Default
LOCAL	whether an Adabas nucleus or Adabas Review hub is isolated and available for local use only.	YES NO	YES

Defines an isolated Adabas nucleus or Adabas Review hub that is only available locally. The nucleus or hub is unknown to Entire Net-Work. A nucleus or Adabas Review hub specifying LOCAL=YES (the default) can have the same database ID or hub ID as another database nucleus or Review hub on another network node.

Value Meaning

YES Isolates this nucleus or Adabas Review hub (that is, makes it unaddressable) from other Entire Net-Work nodes.

NO Allows the nucleus or Adabas Review hub to receive calls from other Entire Net-Work nodes.

Example

In the following example, the Adabas nucleus is isolated and cannot be addressed by other Entire Net-Work nodes.

```
ADARUN PROG=ADANUC,LOCAL=YES
```

NAB : Number of Attached Buffers

Parameter	Specify . . .	Minimum	Maximum	Default
NAB	the number of attached buffers to be used.	1	varies, depending on the amount of available virtual storage	16

The NAB parameter defines the number of attached buffers to be used during the session. An attached buffer is an internal buffer used for interregion communication. It is required in all environments. Adabas allocates an attached buffer pool with a size equal to the value of NAB multiplied by 4096 bytes.

You may specify as many attached buffers as fit into the available virtual storage.

In environments running in 31-bit addressing mode, the attached buffer pool space is allocated above the 16-MB line.

Users of the Adabas Review hub should read [Storage Requirements](#) in *Adabas Review Concepts and Facilities Manual* for more information about the space requirements of the Command Queue for Adabas Review.

The NAB parameter syntax is:

```
NAB={ n | 16 }
```

Specific Product Recommendations

- For Event Replicator Server databases, set parameter NAB to a value greater than or equal to:

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

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

- If data is sent through Entire Net-Work from one or more Adabas nuclei to an Event Replicator Server, the Entire Net-Work NAB parameter must also be set to a value greater than or equal to:

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

Example

The following example runs the Adabas Review hub nucleus with 50 attached buffers.

```
ADARUN PROG=ADAREV,NAB=50
```

NC : Number of Command Queue Elements

Parameter	Specify . . .	Minimum	Maximum	Default
NC	the maximum number of command queue elements.	20	32767	200

The number of command queue elements (CQEs) established for the Adabas or Review hub session determines the maximum number of Adabas commands that may be queued or be in process at any one time during the session.

Each call from the Adabas nucleus is assigned a CQE. The CQE is released when the user receives the results of the command, the Adabas Review hub has processed the command, or the user has been timed out..

192 bytes are required for each CQE.

Software AG recommends that you set NC high enough to allow one command per active user for possible synchronization during execution of the online SAVE database function of the ADASAV utility.

The Adabas session statistics or Adabas Online System can be used to tune this parameter for the next session.

For more information about the space requirements of the Command Queue for Adabas Review, refer to [Storage Requirements](#) in *Adabas Review Concepts and Facilities Manual*.

Example:

Run the Adabas nucleus with a maximum of 500 elements in the command queue.

```
ADARUN PROG=ADANUC,NC=500
```

The following example runs the Adabas Review hub nucleus with a maximum of 500 elements in the command queue.

```
ADARUN PROG=ADAREV,NC=500
```

REVFILTER : Review Record Filtering Control

Parameter	Specify . . .	Possible Values	Default
REVFILTER	whether to allow Adabas Review record filtering during the session.	YES NO	YES

REVFILTER determines whether command log record filtering may be activated. Filtering can decrease the number of command log records passed to Review for report processing.

Value Meaning

YES The default setting. Database command log records may be filtered from report processing, depending upon Review report rules.

NO All command log records will be passed to Review for report processing.

Examples

In the following example, Adabas Review's record filtering may be activated during the Adabas nucleus session.

```
ADARUN PROG=ADANUC,REVFILTER=YES
```

In the following example, Adabas Review's record filtering will not be in effect for the Adabas nucleus session.

```
ADARUN PROG=ADANUC,REVFILTER=NO
```

REVLOGBMAX: Logged Buffer Size Limit for Review

Parameter	Use	Values	Default
REVLOGBMAX	Specify the maximum allowable number of bytes of a logged buffer for Review.	Any integer ranging from 0 to 30000	5120

If a buffer is longer than this value, the logged buffer is truncated from the point at which its size exceeds the setting of the REVLOGBMAX parameter. The REVLOGBMAX setting affects the ADARUN LOGGING parameter specifications only for both CLOGLAYOUT=8. The minimum value (368) is the size of the length of the CLOGLAYOUT=8 basic record plus the length of the extended Adabas control block (ACBX).

Example

The following example runs the Adabas nucleus using a logged buffer size limit of 512. Individual logged buffers, such as the format buffer for an Adabas command, will be truncated if they exceed 512 bytes.

```
ADARUN PROG=ADAREV, LOGBMAX=512
```

REVLOGMAX: Total Logged Buffer Size Limit for a Review Command

Parameter	Use	Values	Default
REVLOGMAX	Specify the maximum size of all of the logged buffers allowed for an Adabas Review command.	Any integer ranging from 2000 to 32768 (32K).	16384

When the sum of sizes of the logged buffers for an Adabas Review command reaches the value of the REVLOGMAX parameter, the buffer exceeding the limit is truncated and all following buffers are omitted. The size of REVLOGMAX must at least be as large as the REVLOGBMAX + 2000 in order to accommodate the largest buffer of the Adabas command. The minimum value (2000) is the size of the length of the CLOGLAYOUT=8 basic record plus the length of the extended Adabas control block (ACBX) and the CLEX information.

The CLOGMAX setting affects the ADARUN LOGGING parameter specifications only for CLOGLAYOUT=8.

Example

The following example runs the Adabas nucleus using a command logged buffer size limit of 32000 bytes. The sum of all logged buffers for an Adabas Review command to this nucleus cannot exceed 10000 bytes.

```
ADARUN PROG=ADAREV, REVLOGMAX=10000
```

SVC : SVC Number

This parameter applies to the operating environments z/OS and z/VSE only.

Parameter	Specify . . .	Possible Values	Default
<u>SVC</u>	the Adabas SVC number or Adabas Review hub SVC number to be used for the session.	see text	45 (z/VSE) 249 (z/OS)

The SVC number is specified as an integer. It must correspond to the number used for the Adabas SVC at your installation.

The Adabas SVC or Adabas Review hub SVC are used to perform various Adabas internal functions under z/OS and z/VSE.

Valid SVC values are as follows:

z/OS 200-255

z/VSE 45 is recommended; any free SVC value can be used. See the Adabas Installation documentation for information about finding free values for z/VSE.

Example

The following example runs an Adabas session under z/OS using SVC 202 for the Adabas SVC.

```
ADARUN PROG=ADANUC ,SVC=202
```

The following example runs an Adabas Review hub session under z/VSE using SVC 45 for the Adabas Review hub SVC.

```
ADARUN PROG=ADAREV ,SVC=45
```

8 Operator Commands

- Entering Operator Commands 70
- Operator Command Overview 70



Note: The commands described in this section of the documentation are used only in hub mode.

The operator commands in this section of the documentation are used to control Adabas Review (ADAREV). They perform the following general types of operations:

- Terminate an Adabas or user session;
- Display nucleus or utility information;
- Log commands into CLOG;
- Change Adabas operating parameters or conditions.

Entering Operator Commands

▶ To enter ADAREV operator commands in z/VSE environments

- 1 Enter an MSG command for the z/VSE partition in which Adabas is executing (for example, MSG F*n*).

Adabas responds when ready for communication with Adabas message ADAI29. In cases where the commands ADAEND and CANCEL cause Adabas to end the nucleus session, no outstanding reply is presented so that an orderly shutdown can occur without the need for operator intervention.

- 2 Enter the desired Adabas operator command or commands.

More than one command can be entered in a session. You can also enter a command followed directly by a slash (/) to end any further operator communications until the next MSG command by z/VSE.

- 3 When all desired commands have been entered, close the operator communication session by entering a null command (EOB).
- 4 To enter any more commands after the session is closed, repeat this entire procedure.

Operator Command Overview

The remainder of this section of the documentation describes the commands that an Adabas Review operator can enter from the console.

ADAEND

Used to terminate an ADAREV session normally; the Adabas Review nucleus is terminated normally. No new monitoring commands are accepted and all currently queued requests are dropped.

CANCEL

Used to terminate ADAREV immediately; the Adabas Review nucleus is abnormally terminated and the job aborts with user completion code 253.

DCLIENT

```
DCLIENT={ dbid | ALL }
```

Displays information on the specified client or ALL clients. The DCLIENT displays the number of clients currently registered with the hub plus the individual status of each client, including the client's DBID, time of last activity, whether the client has any active reports and if buffers are required from the client, the number of reports (and of these, how many reports require buffer information), and the total number of monitoring data records received from the client. The following is an example of the message output:

```
REXH13 hub-id 001 CLIENT(S) LAST-ACT RPT BUF PRS WBUF LOG-RECORDS
REXH13 hub-id 196 03:22:48 Y N 003 000 10,323
```

DCQ

Used to display the entire list of queued requests. The DCQ displays the sequence number, client's job name, client's user ID, request code, and status flags for each queued request.



Note: If a large value was set for NC (as is recommended), the DCQ request may incur delays in the Adabas Review hub processing if a large number of queue elements must be displayed. Also, the display on the operator console may fill the console's buffers causing further system delays.

The following is an example of the message output:

```
AREV07 hub-id 0000000013 NEXT EXPECTED SEQUENCE NUMBER
AREV07 hub-id 0000000011 ADASMP ARVU D (C1D9E5E400C40000) PC 2800
AREV07 hub-id 0000000012 ADASMP ARVU D (C1D9E5E400C40000) PC 2800
```

DNC

Used to display the number of queued requests currently in the command queue.

STARTCLIENT

```
STARTCLIENT={dbid | ALL}
```

Used to initiate a change order command to the specified client or "ALL" clients informing the client(s) to begin sending monitoring data to the hub. The change order is only sent to registered clients (clients that appear on the DCLIENT operator command display).



Note: A change order changes a client's operation only if the monitoring status has been changed. This occurs only in cases where a previous STOPCLIENT operator command had been issued.

STOPCLIENT

```
STOPCLIENT={dbid | ALL}
```

Initiate a change order command to the specified client or ALL clients informing the client(s) to stop sending monitoring data. The change order is only sent to registered clients (clients that appear on the DCLIENT operator command display).

Adabas Operator Commands

The following operator commands can be entered to monitor and control Adabas nucleus operation.

ADAEND

Terminates Adabas session normally. No new users are accepted after this command has been issued. ET logic updating is continued until the end of the current logical transaction for each user. After all activity has been completed as described above, the Adabas session is terminated.

CANCEL

Terminates Adabas session immediately. All command processing is immediately suspended. A pending autorestart is in effect, which in turn causes the autorestart routine to be executed during the initialization of the next Adabas session.

Index

A

- ABEND processing, 41
- Adabas, 72
 - adding startup statements, 42
- Adabas Native SQL
 - z/VSE installation, 24
- Adabas Review
 - accessing, 34
 - allocating storage for, 4
 - operations under z/VSE, 39
 - parameter to set, 60
 - starting under z/VSE, 29
- Adabas Review Data Communication, 3
- ADAEND operator command
 - terminate Adabas session normally, 72
 - terminate ADAREV session normally, 71
- ADAREV operator commands, 70
 - ADAEND, 71
 - CANCEL, 71
 - DCLIENT, 71
 - DCQ, 71
 - DNC, 72
 - STARTCLIENT, 72
 - STOPCLIENT, 72
- ADARUN parameters
 - Logged Buffer Size Limit for Review, 66
 - NAB to size the attached buffer, 5-6
 - NC to size the command queue, 5
 - REVLOGBMAX parameter, 66
 - REVLOGMAX parameter, 67
 - syntax, 58
 - Total Logged Buffer Size Limit for a Review Command, 67
 - under z/VSE, 57
- attached buffer, 6
 - parameter to set time limit for hold, 61
- attached buffers
 - number of
 - parameter to specify, 64

B

- batch
 - z/VSE installation, 27
- buffer pool
 - attached
 - space allocation, 64
- BUFFER-SEGMENTS parameter, 7

C

- CANCEL operator command
 - terminate Adabas session immediately, 72
 - terminate ADAREV session immediately, 71
- CICS
 - supported levels, 4
 - z/VSE installation, 26
- CLOSE-DBID parameter
 - CONFIGDB file, 53
- Com-plete
 - supported levels, 4
 - z/VSE installation, 26
- command logging
 - considerations, 48
 - setting up, 48
 - user exit, 49
- command logging files, 43
- command queue, 5
- command queue element
 - maximum number of
 - parameter to specify, 65
 - parameter to set time limit for hold, 61
- commands
 - EDIT CONFIGDB, 51
 - hub mode operator, 70
 - INSTALL, 30-31
 - parameter to
 - set time limit for completion, 61
- CONFIGDB file
 - modifying, 51
 - parameter descriptions, 52
- configuration parameters
 - modifying, 51
- CT
 - ADARUN parameter, 61
- CURSOR-POSITION parameter
 - CONFIGDB file, 52

D

- data file
 - initializing, 31
- DCLIENT operator command
 - display client information, 71
- DCQ operator command
 - display list of queued requests, 71
- DECIMAL-CHAR parameter
 - CONFIGDB file, 52

DNC operator command
display number of requests in command queue, 72

E

EDIT CONFIGDB command, 51
environments supported, v

F

files
command logging, 43
summary logging, 43
FORCE
ADARUN parameter, 62

H

hub mode
operator commands, 70
hub server
attached buffer storage requirements, 6
command queue storage requirements, 5
storage requirements, 4-5

I

ID Table
parameter to
allow nucleus to overwrite existing entry, 62
initialization
process screen, 33
process window, 31
INSTALL command, 30-31
Installation
prerequisites, 4
installation
Adabas Review repository, 17
adding Adabas startup statements in local mode, 42
hub mode overview, 11
non-TP-specific components in local mode, 14
overview, 11
TP-specific components, 25
verifying, 35
verifying under CICS, 36
verifying under Com-plete, 35
z/VSE, 3
z/VSE Adabas Native SQL support, 24
z/VSE support for reporting from batch Natural, 24
z/VSE under Adabas, 17-18
z/VSE under CICS, 26
z/VSE under Com-plete, 26
z/VSE under Natural, 15

L

LOCAL
ADARUN parameter, 63
local mode
defined, v
storage requirements, 9
LOGUEXIT sample copy job, 51
LOGUEXIT sample user exit, 49, 51

M

MAXIMUM-MAXK parameter
CONFIGDB file, 53
MAXSTORE
report parameter, 8

N

NAB
ADARUN parameter, 64
NATPARM
parameter settings, 16
NATURAL
NTFILE definition, 16
Natural
required parameter settings, 16
user exits, 55
z/VSE support for reporting from batch Natural, 24
Natural Review, 3
NC
ADARUN parameter, 65
nucleus
isolated
parameter to define as a local nucleus, 63
SVC for
parameter to specify, 67
nucleus operator commands, 72

O

OPEN-DBID parameter
CONFIGDB file, 53
operator commands
entering under z/VSE, 70

P

P-UEXITn, 55
PC-FILE parameter
CONFIGDB file, 52
PROGRAM
ADARUN parameter, 59

R

reports
MAXSTORE parameter, 8
storage for control blocks, 8
storage for data collection areas, 8
storage requirements, 4, 8
total storage limit (MAXSTORE), 8
REVFILTER
ADARUN parameter, 66
REVIEW
ADARUN parameter, 60
REVIEW-BUFFER
storage requirements, 4, 7
REVIEWDB-UEx parameter
CONFIGDB file, 53
REVLOGBMAX parameter, 66-67
RVLOGxx command logging files, 43
RVSUMxx summary logging files, 43
RVUALT

- alternate history file, 44
- RVUAUT1 and 2
 - autostarted report definition data sets, 45
- RVUCARD
 - data set for GENCARD-created parameters, 45
- RVUEXI
 - editing the parameters, 47
 - operating environment parameter file, 45
 - timeout parameters, 47
 - user-specified parameters, 47
- RVUEXP
 - error file for RVUEXI, 45
- RVUFLD
 - parameter file, 45
- RVUPARM
 - dummy data set for compatibility, 46
- RVUPRTnn
 - logical printer files, 46
- RVXB-MESSAGE parameter
 - CONFIGDB file, 52

S

- session
 - SVC for
 - parameter to specify, 67
- STARTCLIENT operator command
 - instruct client to begin sending data to hub, 72
- STOPCLIENT operator command
 - instruct client to stop sending data to hub, 72
- Storage
 - allocating above the 16MB line, 4
- storage
 - MAXSTORE report parameter, 8
 - requirements, 4
- summary logging
 - considerations, 49
 - setting up, 50
 - user exit, 50
- summary logging files, 43
- SVC
 - ADARUN parameter, 67
- System Maintenance Aid, 4

T

- target
 - default definition window, 32
- timeout control
 - interregion communication limit
 - parameter to set, 61
- TP monitors
 - supported for OS/390, 4

U

- UBAR parameter
 - CONFIGDB file, 52
- UCMD-TIMEOUT parameter
 - RVUEXI operating environment, 47
- UEX5, 53
- UIDT-CELLS parameter
 - RVUEXI operating environment, 47
- UIDT-TIMEOUT parameter

- RVUEXI operating environment, 47
- user exits
 - exit 5, 53
 - hub event handler, 53
 - Natural, 55
- user profile system, 3
 - initializing, 30
- users
 - storage requirements for online use, 4, 8

W

- work areas
 - storage requirements, 4, 9

Z

- z/VSE
 - entering operator commands, 70
 - installation, 3
 - installation of support for reporting from batch Natural, 24
 - installation under Adabas, 17-18
 - installation under batch, 27
 - installation under CICS, 26
 - installation under Com-plete, 26
 - installation under Natural, 15
 - installing Adabas Native SQL support, 24
 - repository installation, 17

