Adabas Review

Installation and Operations for BS2000

Version 4.5.1

June 2014

Adabas Review

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.

Document ID: REV-BS2000-INSTALL-451-20140626

Table of Contents

Installation and Operations for BS2000 v
1 Conventions 1
2 Preparing for the Installation 3
- Prerequisites 4
- Storage Requirements 5
- The Adabas Review Installation Tape 9
- Installation Overview 10

3 Installation: Phase 1 11
- Copy the Tape Contents to Disk 12
- Install Adabas Review under Natural 14
- Install the Adabas Review Repository 16
- Install Adabas Review under Adabas 17
- Install the Adabas Review Hub 24
- Optional Installation Procedures 25

4 Installation: Phase 2 27
- Step 1. Install REVEXITB for the Adabas Link Routines 28
- Step 2. Set the Adabas Entry Name, if necessary 28
- Step 3. Link the Natural Front-End, if necessary 29
- Step 4. Modify the Natural Startup Procedure, if applicable 29

5 Starting Adabas Review 31
- Starting Adabas Review for the First Time 32
- Accessing Adabas Review 36
- Verifying the Installation Under TIAM/Batch or UTM 37

6 Operations 39
- SYSOUT and a Subtask List Files 40
- Files Used by Adabas Review 41
- Editing the RVUEXI Parameter File 44
- Command Logging Considerations 45
- Summary Logging Considerations 47
- Modifying Configuration Parameters 49
- User Exit 5 (Adabas Review Hub Event Handler) 51
- Adabas Review Natural User Exits 53

7 ADARUN Parameters for Adabas Review in BS2000 55
- PROGRAM : Program to Run 56
- REVIEW : Adabas Review Control 57
- CMDQMODE : Command Queue Mode 58
- FORCE : Allow Nucleus Database ID or Review Hub Table Entry Overwrite 59
- NAB : Number of Attached Buffers 60
- NC : Number of Command Queue Elements 61
- REVFILTER : Review Record Filtering Control 62
- SUBMPSZ : GETMAIN Memory Pool for Subtasks 63

8 Operator Commands 65
- Entering Operator Commands 66

Operator Command Overview 66
Index 69

Installation and Operations for BS2000

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.

Note that Adabas Review version 4.1 or later is required to monitor Adabas version 8.0 or later databases.

The Adabas Review installation documentation for BS2000 is organized as follows:

Preparing for the Installation	Provides the information needed to prepare for the installation of Adabas Review under BS2000 operating system environments under the teleprocessing monitors TIAM/batch or UTM.
Installation Phase 1	Describes the first phase of the Adabas Review installation process. Phase 1 comprises all the steps that are independent of any particular TP monitor environment.
Installation Phase 2	Describes the second phase of the Adabas Review installation process. Phase 2 installs the components that are specific to the particular TP monitor in use at your site.
Starting Adabas Review	Discusses the initialization procedures performed by Adabas Review during startup.
Operations	Describes operation procedures for Adabas Review.
ADARUN Parameters for Adabas Review in BS2000	Discusses the Adabas ADARUN initialization parameters, most for use with the hub only, in BS2000 environments.
Operator Commands	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 (file) or module names.

Placeholder	Meaning	Definition
v	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.
r	release	**Minor Version** The second digit of the version number indicates functionality addition or enhancement that adds value to the product.
s	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 5
- The Adabas Review Installation Tape 9
- Installation Overview 10

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

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

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

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

For installation with SMA please read #READ-ME.REV.

Prerequisites

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

Operating System Level

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

TP Monitor Support

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

- TIAM
- UTM

Storage Requirements

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

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

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

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

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

Storage for the Hub

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

Two queues, both controlled by the database administrator (DBA), are used by the Adabas Review hub: the command queue (sized using the ADARUN parameter `NC`) and the attached buffer (sized using the ADARUN parameter `NAB`). For more information, read about these ADARUN parameters in *ADARUN Parameters*, 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 BUFFERS-32K parameter. If this parameter is not specified, any BUFFERS-32K parameter setting is ignored. If a value below the minimum value is specified for this parameter, the default is used.	124	256
BUFFERS-32K	The number of buffer pool entries that have a length of 4097 bytes or greater. This parameter is usually specified along with the BUFFERS-4K	15	30

Parameter	Description	Minimum Value	Default Value
	parameter. If this parameter is not specified, any BUFFERS-4K parameter setting is ignored. If a value below the minimum value is specified for this parameter, the default is used.		

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.

The Adabas Review Installation Tape

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

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

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

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

Installation Overview

Adabas Review is installed in two phases:

1. Phase 1: Install non-TP-specific components. All steps in this phase are identical regardless of the TP monitor in use:
   - Install Adabas Review under Natural;
   - Install the Adabas Review repository;
   - Install Adabas Review under Adabas;
   - Install the Adabas Review hub (hub mode only);
   - Optional installation procedures.
2. Phase 2: Install TP-specific components. Separate procedures are used to install Adabas Review under each of the supported TP monitors: TIAM, batch, and UTM.

Phase one procedures are described in *Installation: Phase 1*, elsewhere in this guide; phase two procedures in *Installation: Phase 2*, elsewhere in this guide.

3 Installation: Phase 1

- Copy the Tape Contents to Disk 12
- Install Adabas Review under Natural 14
- Install the Adabas Review Repository 16
- Install Adabas Review under Adabas 17
- Install the Adabas Review Hub 24
- Optional Installation Procedures 25

This section of the documentation provides installation procedures for Adabas Review in a BS2000 operating system environment. These procedures must be completed by all BS2000 sites installing Adabas Review.

- in local mode as an extension of ADALOG;
- in hub mode as a server in its own address space with client interface modules in the address space of all Adabas databases to be monitored.

Note that Adabas Review version 4.1 or above is required to monitor Adabas version 8.0 and above databases.

Copy the Tape Contents to Disk

The Adabas Review installation tape can be unloaded to disk using one of the procedures described below. In these procedures, the following values must be supplied:

- In the file names, replace *vrs* with the current version, revision, and system maintenance level of the product;
- Replace *xxxxxx* with the volume serial number of the tape.

- Installation Using SMA
- Installation Not Using SMA

Installation Using SMA

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

Installation Not Using SMA

If you are not using SMA, copy the data sets (files) from tape to disk using the procedure described below:

- Step 1: Copy the Library SRVnnn.LIB from Tape to Disk
- Step 2: Copy the Procedure COPY.PROC from Tape to Disk

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

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

Note: This step is not necessary if you have already copied the library SRV*nnn*.LIB from another Software AG tape. For more information, refer to the element #READ-ME in this library.

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

```
/IMPORT-FILE SUPPORT=*TAPE(FILE-NAME=SRVnnn.LIBS, -
/ VOLUME=<volser>, DEV-TYPE=<tape-device>)
/ADD-FILE-LINK LINK-NAME=EDTSAM, FILE-NAME=SRVnnn.LIBS, -
/ SUPPORT=*TAPE(FILE-SEQ=3), ACC-METH=*BY-CAT, -
/ BUF-LEN=*BY-CAT, REC-FORM=*BY-CAT, REC-SIZE=*BY-CAT
/START-EDT
@READ '/'
@SYSTEM 'REMOVE-FILE-LINK EDTSAM'
@SYSTEM 'EXPORT-FILE FILE-NAME=SRVnnn.LIBS'
@WRITE 'SRVnnn.LIBS'
@HALT
/ASS-SYSDTA SRVnnn.LIBS
/MOD-JOB-SW ON=1
/START-PROG $LMS
/MOD-JOB-SW OFF=1
/ASS-SYSDTA *PRIMARY
<tape-device> = device type of the tape, for example, TAPE-C4
<volser> = VOLSER of tape (see Report of Tape Creation)
```

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

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

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

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

Step 3: Copy all Product Files from Tape to Disk

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

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

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

Install Adabas Review under Natural

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

- Step 1. INPL Adabas Review
- Step 2. Modify the Natural Parameter File
- Step 3. Define Adabas Review to Natural Security, if applicable

Step 1. INPL Adabas Review

(JOB 1061, step 2600)

INPL the Adabas Review programs and DDMs from the REV*vrs*.INPL file to your Natural system files.

You can use any of your site-dependent Natural INPL JCL.

The Natural programs are copied into the SYSREVDB library in your FNAT file.

Step 2. Modify the Natural Parameter File

(Job I060 batch; Job I080 online)

To modify the Natural parameter file, complete the following steps:

1 Include the minimum following parameter settings in the Natural parameter module that will be used when accessing Adabas Review:

```
LS=250
PS=80
MADIO=5000
MAXCL=0
ESIZE=40
NTWORK(7),AM=PC
```

NTWORK 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.

2 Include the following parameter in *all* your installation Natural parameter modules: (If this is not done, *no* data relating to Natural will be reported on.)

```
ADAPRM=ON
```

3 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.

4 Reassemble and link the *NATPARM* module to your Natural nucleus.

Step 3. Define Adabas Review to Natural Security, if applicable

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

If Natural Security is installed, complete the following steps:

1 Define the SYSREVDB library for the Adabas Review system.

2 Define the following Adabas Review DDMs to Natural Security as public DDMs:

```
REVIEW-Adabas-Vvrs-CLOG
REVIEW-FNAT
REVIEW-Adabas-Vvrs-SYSTEM
```

Install the Adabas Review Repository

(JOB I050, step 2600)

This procedure applies to both local and hub mode installations.

The Adabas Review repository is a system file used for storing user profiles and descriptions of interactive reports, target definitions, and for saving historical data accumulated by Adabas Review reports. Any 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 to a version 4.5 repository, run the EXPAND3 job from the source library.

This section covers the following topics:

- Step 1. Modify the Sample JCL for Loading the Adabas Review Repository
- Step 2. Run the Modified Job

Step 1. Modify the Sample JCL for Loading the Adabas Review Repository

You can use any of your site-dependent ADALOD JCL to load the FDT from the file REV*vrs*.SYSF using the following ADALOD parameters:

```
ADALOD LOAD FILE=fnr
ADALOD NAME=REVvrs-DBFILE
ADALOD VERSION=5
ADALOD ASSOPFAC=10
ADALOD DATAPFAC=10
ADALOD DSSIZE=5
ADALOD NISIZE=500B
ADALOD MAXISN=10000
ADALOD NUMREC=3000
ADALOD SORTSIZE=5
ADALOD TEMPSIZE=5
```

Before submitting the job, change:

- the ADALOD LOAD FILE=*fnr* statement to reflect the number of the file that will contain the Adabas Review repository.
- (hub mode) 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.

Step 2. Run the Modified Job

Use the job you modified in the previous step to load the Adabas Review repository file, JOB I050, step 2600.

Install Adabas Review under Adabas

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

- Step 1. Modify the Adabas Initialization Parameters
- Step 2. Modify the Adabas Review AUTO-START Generator Job
- Step 3. Modify the Adabas Startup JCL
- Step 4. Apply Adabas ZAPs

Step 1. Modify the Adabas Initialization Parameters

Modify your existing Adabas ADARUN parameters to include the following (these ADARUN cards are generated by SMA):

For local mode only:

```
ADARUN REVIEW=LOCAL
```

For hub mode only:

The Adabas Review hub ID value is set using the ADARUN parameter `REVIEW`.

```
ADARUN REVIEW=hubid
ADARUN UEX5=user-exit
```

where *hubid* is the Adabas Review hub ID (version 4.2 and above support two-byte DBIDs) and UEX5 is optional as described in the section ***User Exit 5 (Adabas Review Hub Event Handler)*** in the section ***Operations***, elsewhere in this guide.

Notes:

1. Adabas Review does not require you to log any of the Adabas buffers. You can select options when writing data in parallel to the CLOG.
2. In local mode, physical command logging can be suppressed by specifying the LOGGING=NO option on the Adabas Review INPUT statement or target definition for the file. The default value for the LOGGING parameter is LOGGING=NO.

Step 2. Modify the Adabas Review AUTO-START Generator Job

Note: This step is not supported by the automated SMA installation.

The AUTO-START generator job is started automatically by Adabas Review when:

- the target definition for the relevant database changes;
- the AUTOSTART option in an existing report definition changes;
- a new report is defined with the option AUTOSTART=Y;
- the GENAUTO or GA command is entered by the user; or
- the GENCARD or GC command is entered by the user.

Note: The AUTO-START generate process is a BS2000 subtask. This is a spawned task on BS2000. The output for this task is found in the files L.O.*tttt*.RAOSAUTO.*hex_timestamp* where:

tttt	is the task number of the Adabas Review nucleus or hub;
hex_timestamp	is the STCK value of the clock in hex when the subtask was spawned.

This section covers the following topics:

- Modify the P.GENERATE job
- About the Adabas Review History File Population Job

Modify the P.GENERATE job

The base P.GENERATE job is located in the REV*vrs*.SRC data set (file) and should be modified appropriately for your site.

Each database running with Adabas Review and the hub needs a set of files and jobs for itself. P.GENERATE generates the files described in the rest of this section to satisfy this need.

Data files:

```
prefix.REVvrs.RVUALT
prefix.REVIEW.RVUAUT1
prefix.REVIEW.RVUAUT2
prefix.REVIEW.RVUPARM
prefix.REVIEW.RVLOG01
prefix.REVIEW.RVLOG02
prefix.REVIEW.RVSUM01
prefix.REVIEW.RVSUM02
prefix.REVIEW.RVUALT.BACKUP
prefix.REVIEW.RVUFLD
prefix.REVIEW.RVUPRT00
prefix.REVIEW.RVUPRT01
prefix.REVIEW.RVUPRT02
prefix.REVIEW.RVUPRT03
prefix.REVIEW.RVUPRT04
prefix.REVIEW.RVUPRT05
prefix.REVIEW.RVUPRT06
prefix.REVIEW.RVUPRT07
prefix.REVIEW.RVUPRT08
prefix.REVIEW.RVUPRT09
prefix.REVIEW.RVUEXP
prefix.REVIEW.RVUEXI
prefix.REVIEW.RVUCARD
```

Editor copy files:

```
REVIEW.COPY.db
```

Call/enter procedures:

```
REVIEW.E.COPYLOG1.db
REVIEW.E.COPYLOG2.db
REVIEW.P.COPYALT.db
```

where:

Item	P.GENERATE Parameter	Description
	&ADAL	the name of the Adabas module library
db	&DB	Adabas Review database number (local or hub: 5 digits with leading zeros)
	&FUNC	<blank> - help information:
		CRE create files DEL delete files
	&IDTNAME	8-character IDT name where the Adabas Review database or hub is to run (must begin with "ADA").

Item	P.GENERATE Parameter	Description
	&LOGPFX	RVLOG. This is the prefix of the logging file.
	&LOGSIZE	If "0", it operates as in Adabas Review 4.4 and the log file will be continuously written to. If a value greater than zero (>0), a file with no secondary extent will be created. When this is full, the next RVLOG (*nn*+1) will be opened and written to.
prefix	&PREFIX	a prefix to the files for easy maintenance (in this documentation, the example prefix is DB099)
	&REPDB	database number (5 digits with leading zeros) of the Adabas Review repository file where the REV*vrs*.SYSF file was loaded.
REV*vrs*	&REV	the current Adabas Review version, revision, and system maintenance level
	&REVL	the name of the Adabas Review module library
	&SUMFX	RVSUM. This is the prefix of the summary file.
	&SUMSIZE	If "0", it operates as in Adabas Review 4.4 and the summary file will be continuously written to. If a value greater than zero (>0), a file with no secondary extent will be created. When this is full, the next RVSUM (*nn*+1) will be opened and written to.
Note: Only two RVLOG and RVSUM files are included in this example procedure. More can be added (RVLOG03 or RVSUM03, for example) by copying and pasting the creation commands.		

About the Adabas Review History File Population Job

Adabas Review report definitions specify whether the data accumulated by the report is also written to the Adabas Review repository. The reports can then be viewed again or combined with previous occurrences of the same report to produce a combined summary report.

Because Adabas Review cannot write to the Adabas Review repository directly when running in local mode, it writes an intermediate history file either at intervals or at Adabas termination. At the next Adabas startup, Adabas Review reads the file with the link name RVUALT and populates its system file with history data.

Note: The Adabas Review history file population process is a BS2000 subtask. This is a spawned task on BS2000. The output for this task is found in the files L.O.*tttt*.RAOSHIST.*hex_timestamp* where

tttt	is the task number of the Adabas Review nucleus or hub;
hex_timestamp	is the STCK value of the clock in hex when the subtask was spawned.

Step 3. Modify the Adabas Startup JCL

This section describes how to modify the Adabas local mode startup JCL and the hub mode client interface JCL.

- Modifying the local mode JCL
- Modifying the hub mode client interface JCL

Modifying the local mode JCL

An example Adabas Review local mode startup job is provided in REV*vrs*.SRC(LOCAL).

Note: The example job LOCAL in the REV*vrs*.SRC library on BS2000 does not set the DBID parameter because we believe that setting a default DBID in this job might lead to problems in the setup of the environment. Instead, the DBID should be specified in the Natural area of Adabas Review for the default DBID in the user profile, or the Natural LFILE=(241,...) parameter. And for the Adabas Review hub database, the REVIEW parameter should set the hub DBID (or if is LOCAL, then it is its own DBID).

Modify the Adabas startup JCL as follows (these modifications are generated by SMA):

1 Add the following step after the nucleus has been terminated to copy the RVUALT file to a backup file that is read by RAOSHIST:

```
/SET-JOB-STEP
/CALL-PROC REVIEW.P.COPYALT.db
```

where REVIEW.P.COPYALT.*db* was generated by P.GENERATE above.

2 Add the FILE/LINK statements required for Adabas Review. Use the file REVIEW.COPY.*db* that was generated by P.GENERATE above.

If for some reason you do not want to use the generated file, you can add the following cards to the deck instead:

```
/SET-FILE-LINK  RVUALT,DB099.REVIEW.RVUALT
/SET-FILE-LINK  RVUAUT1,DB099.REVIEW.RVUAUT1
/SET-FILE-LINK  RVUAUT2,DB099.REVIEW.RVUAUT2
/SET-FILE-LINK  RVUPARM,*DUMMY
/SET-FILE-LINK  RVLOG01,DB099.REVIEW.RVLOG01
/SET-FILE-LINK  RVLOG02,DB099.REVIEW.RVLOG02
/SET-FILE-LINK  RVSUM01,DB099.REVIEW.RVSUM01
/SET-FILE-LINK  RVSUM02,DB099.REVIEW.RVSUM02
/SET-FILE-LINK  RVUFLD,DB099.REVIEW.RVUFLD
/SET-FILE-LINK  RVUPRT00,DB099.REVIEW.RVUPRT00
/SET-FILE-LINK  RVUPRT01,DB099.REVIEW.RVUPRT01
/SET-FILE-LINK  RVUPRT02,DB099.REVIEW.RVUPRT02
/SET-FILE-LINK  RVUPRT03,DB099.REVIEW.RVUPRT03
```

```
/SET-FILE-LINK  RVUPRT04,DB099.REVIEW.RVUPRT04
/SET-FILE-LINK  RVUPRT05,DB099.REVIEW.RVUPRT05
/SET-FILE-LINK  RVUPRT06,DB099.REVIEW.RVUPRT06
/SET-FILE-LINK  RVUPRT07,DB099.REVIEW.RVUPRT07
/SET-FILE-LINK  RVUPRT08,DB099.REVIEW.RVUPRT08
/SET-FILE-LINK  RVUPRT09,DB099.REVIEW.RVUPRT09
/SET-FILE-LINK  RVUEXP,DB099.REVIEW.RVUEXP
/SET-FILE-LINK  RVUEXI,DB099.REVIEW.RVUEXI
/SET-FILE-LINK  RVUCARD,DB099.REVIEW.RVUCARD
```

Notes:

1. Use the supplied procedure REV*vrs*.SRC(P.GENERATE) to generate the required files for each database where Adabas Review (local or hub) is to be installed. Otherwise, I/O errors will be returned during Adabas Review initialization.
2. All databases that have Adabas Review (local or hub) installed must be assigned their own set of Adabas Review files.
3. *DB099* is the `&PREFIX` parameter of the P.GENERATE procedure above.

3 Set up a sequential file with the following contents:

```
ADALNK IDTNAME=idtname
```

where *idtname* is an 8-character IDT name where the Adabas Review is running, the same as in the ADARUN cards. The file *must* be a permanent file because it supplies routing and buffer information to the subtask.

Then set the following link card in the nucleus startup job

```
/SET-FILE-LINK DDLNKPAR,ddlnkpar_file_name
```

where *ddlnkpar_file_name* is the file name of the permanent sequential file above.

4 Ensure that the REVB*vrs*.MOD library is accessible from the executed ADARUN job by declaring the following BLSLIB card:

```
/START-PROGRAM (Adabas.MOD,ADARUN),RUN-MODE=ADV (ALT-LIB=YES)
```

Ensure that Adabas can access the Adabas Review default link name and alternative libraries as declared by BLSLIB above by including the following statement:

```
/SET-FILE-LINK ADALIB,ADAvrs.MOD
/SET-FILE-LINK REVLIB,REVvrs.MOD
/SET-FILE-LINK BLSLIB00,REVvrs.MOD
/SET-FILE-LINK BLSLIB01,ADAvrs.MOD
```

Modifying the hub mode client interface JCL

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

Important: If you are currently running Adabas Review DB 3.4 or below, you must completely deinstall it from the Adabas startup JCL.

To modify the Adabas startup JCL for the hub mode client interface:

1 Use the following BLSLIB structure:

```
*/SET-FILE-LINK    DDLIB,Adabas.MOD
/SET-FILE-LINK    REVLIB,REVvrs.MOD
/SET-FILE-LINK    BLSLIB00,REVvrs.MOD
/SET-FILE-LINK    BLSLIB01,ADAvrs.MOD
```

2 Add a DDLOG or DDCLOGR1/DDCLOGR2 file.

This DDLOG FILE card *must* point to a physical output file and *must not* be coded as DUMMY and *not* be allocated as a temporary file.

Note: If Adabas command logging (single or dual) is already being used, this portion of this step can be omitted.

3 Use the following to ensure that Adabas can access alternative libraries as declared by BLSLIB above:

```
START-PROGRAM (Adabas.MOD,ADARUN),RUN-MODE=ADV (ALT-LIB=YES)
```

4 Restart Adabas.

Step 4. Apply Adabas ZAPs

Upgrade the Adabas Review module libraries with Adabas fixes.

Adapt the procedures BS2ZAP and ADA*vrs*.Z0000 in REV*vrs*.ZAPS, then run ADA*vrs*.ZAPS where *vrs* is the version, revision, and system maintenance level of the Adabas Review being installed.

Install the Adabas Review Hub

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

- Step 1. Modify the Adabas Startup JCL (Hub Mode Server Database)
- Step 2. Start the Adabas Review Hub for the First Time

Step 1. Modify the Adabas Startup JCL (Hub Mode Server Database)

An example Adabas Review hub mode startup job is provided in REV*vrs*.SRC(HUBJCL).

Note: The example job HUBJCL in the REV*vrs*.SRC library on BS2000 does not set the DBID parameter because we believe that setting a default DBID in this job might lead to problems in the setup of the environment. Instead, the REVIEW parameter should set the hub DBID for the hub database (or if is LOCAL, then it is its own DBID).

Modify the Adabas startup JCL as follows (these modifications are generated by SMA):

1 Correct any library names or file names.

2 Modify the ADARUN parameter `REVIEW` to reflect the target DBID you will use for the Adabas Review hub.

3 Specify the ADARUN parameter `IDTNAME`, if necessary. This value must be the same as that used by the sending Adabas nuclei.

The ADARUN parameters supplied in the sample JCL member HUBJCL are the only ones recognized for setting up the hub nucleus. See the section *ADARUN Parameters* for information about these ADARUN parameters.

Software AG recommends that you set the dispatching priority of the Adabas Review hub higher than that used by the sending Adabas nuclei.

4 Set up a sequential file with the following contents:

```
ADALNK IDTNAME=idtname
```

where `idtname` is an 8-character IDT name where the Adabas Review is running, the same as in the ADARUN cards. The file *must* be a permanent file because it supplies routing and buffer information to the subtask.

Then set the following link card in the nucleus startup job

```
/SET-FILE-LINK DDLNKPAR,ddlnkpar_file_name
```

where `ddlnkpar_file_name` is the file name of the permanent sequential file above

Step 2. Start the Adabas Review Hub for the First Time

See the section ***Starting Adabas Review***, elsewhere in this chapter.

Optional Installation Procedures

This section describes optional installation procedures

- Implement support for Adabas Native SQL

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.

4 Installation: Phase 2

- Step 1. Install REVEXITB for the Adabas Link Routines 28
- Step 2. Set the Adabas Entry Name, if necessary 28
- Step 3. Link the Natural Front-End, if necessary 29
- Step 4. Modify the Natural Startup Procedure, if applicable 29

REVEXITB for BS2000 is supplied in the module RDBLXBS2 for TIAM, batch, and UTM applications. This module should be linked to ADALNK, as described in the steps in this chapter. A sample job is provided in REV*vrs*.SRC(P.ADALNK).

Note: If you elect to use the link routine ADALNKX, link the batch Adabas Review routine RDBLXBS2 and RDBLKIND to ADALNK ensuring that the symbols ADALNKDA, SSFB2C and ADAL2P are not hidden.

Step 1. Install REVEXITB for the Adabas Link Routines

If you use ADAUSER to load ADALNK, generate a file with the following structure:

-
  ```
  ADALNK IDTNAME=database_IDT
  ```

  where *database_IDT* is the IDT name where the database being monitored by Adabas Review is registered.

  Also, place the following statement in your job:

  ```
  /SET-FILE-LINK DDLNKPAR,adalink_file_name
  ```

If you use ADARUN to load ADALNK, add the following fields to the DDCARD file:

-
  ```
  ADARUN   PROG=USER,.....IDTNAME=database_IDT
  ```

  where *database_IDT* is the IDT name where the database being monitored by Adabas Review is registered.

Step 2. Set the Adabas Entry Name, if necessary

If you have not already set the Adabas entry name, do so now.

In the NATFRONT(batch),NRTFRONT(TIAM) or NUTFRONT(UTM) source modules, set:

```
ADACOM=Adabas
```

Then reassemble the appropriate source module.

Step 3. Link the Natural Front-End, if necessary

(JOB I060) for batch; (JOB I080) for TIAM and UTM.

Step 4. Modify the Natural Startup Procedure, if applicable

Modify your existing Natural startup procedure by including the following /SET-FILE-LINK statement pointing to the Adabas library where your ADALNK module is located:

```
/SET-FILE-LINK DDLIB,Adabas.MOD
```

Note: If you are *not* using a procedure, the above /SET-FILE-LINK statement must be supplied before calling Natural online.

5 Starting Adabas Review

- Starting Adabas Review for the First Time 32
- Accessing Adabas Review 36
- Verifying the Installation Under TIAM/Batch or UTM 37

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

- Initializing Adabas Review for the first time;
- Accessing Adabas Review;
- Verifying the Adabas Review installation.

Starting Adabas Review for the First Time

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

- the user profile subsystem, which allows you to give users access to Adabas Review; and
- the Adabas Review data file, which designates a DBID for the Adabas Review repository, and installs the Adabas Review-supplied reports.

Initializing the Adabas Review User Profile Subsystem

To initialize the Adabas Review user profile subsystem

1. Access Natural as you normally do.
2. At the NEXT prompt, log on to the library 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 `INSTALL UP` and press ENTER.

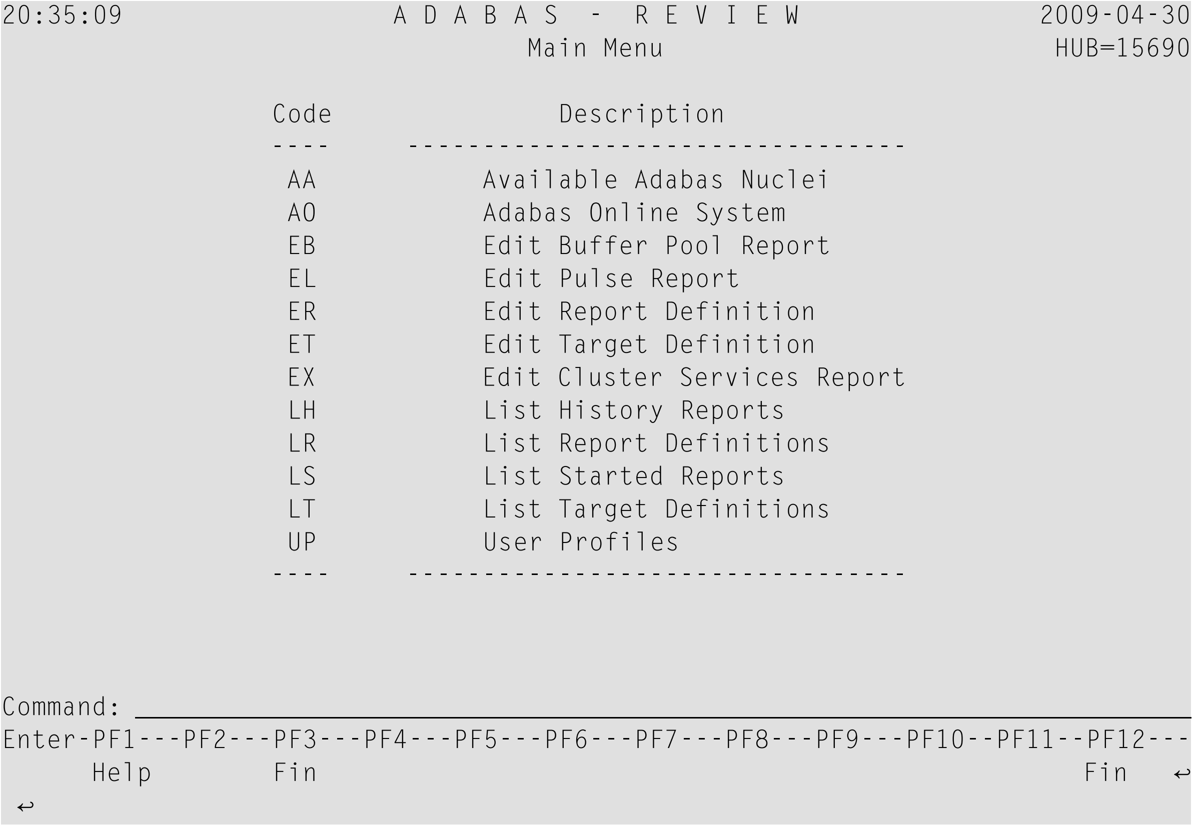
   This initializes the Adabas Review user profile subsystem and adds one default user record to the system. You must have this initial user defined in order to enter Adabas Review.

   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` (use the UP option displayed below and read *Customizing the Default Profile* , in the *Adabas Review Administration Guide* for more information). You can also manually override the default user profile HUB / DBID value by issuing HUB=*hubid* from the command line after you log onto Adabas Review.

4 At the NEXT prompt, type `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   ↩
 ↩
```

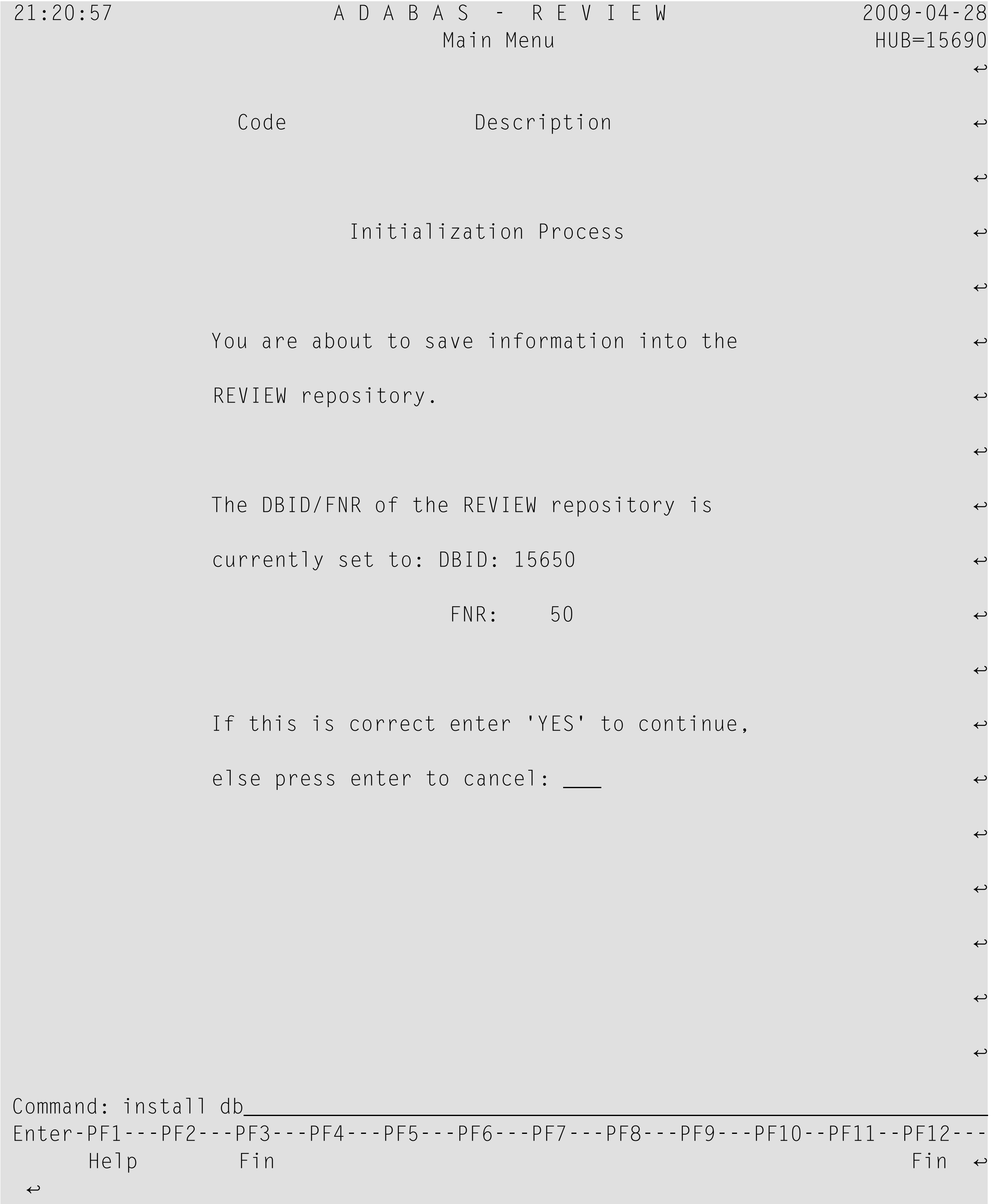
Initializing the Adabas Review Data File

Note: If you are using a repository from version 4.3, omit this step.

To initialize the Adabas Review data file

1 At the Adabas Review main menu command line, type `INSTALL DB` and press ENTER.

The Initialization Process window appears:



```
21:20:57                  A D A B A S  -  R E V I E W                 2009-04-28
                                    Main Menu                          HUB=15690
                                                                               ↩
                  Code                Description                              ↩
                                                                               ↩
                           Initialization Process                              ↩
                                                                               ↩
                You are about to save information into the                     ↩
                REVIEW repository.                                             ↩
                                                                               ↩
                The DBID/FNR of the REVIEW repository is                       ↩
                currently set to: DBID: 15650                                  ↩
                                  FNR:    50                                   ↩
                                                                               ↩
                If this is correct enter 'YES' to continue,                    ↩
                else press enter to cancel: ___                                ↩
                                                                               ↩
                                                                               ↩
                                                                               ↩
                                                                               ↩
                                                                               ↩
Command: install db______________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Fin                                                    Fin  ↩
 ↩
```

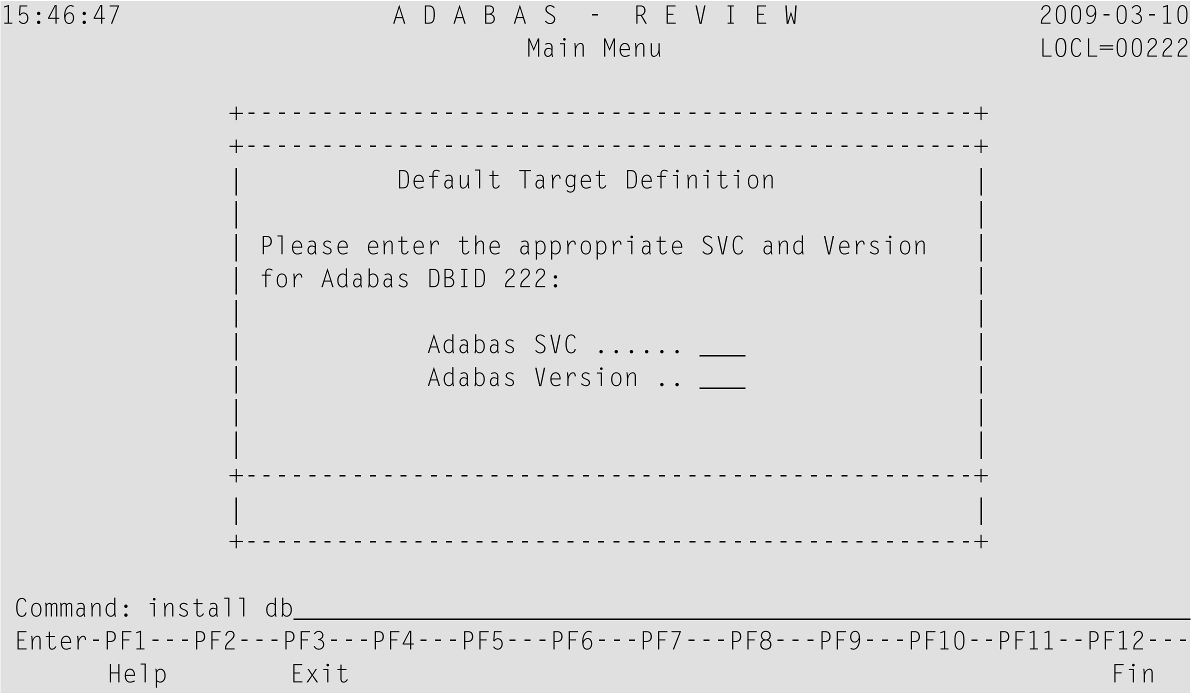
2 Confirm or cancel the initialization.

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.



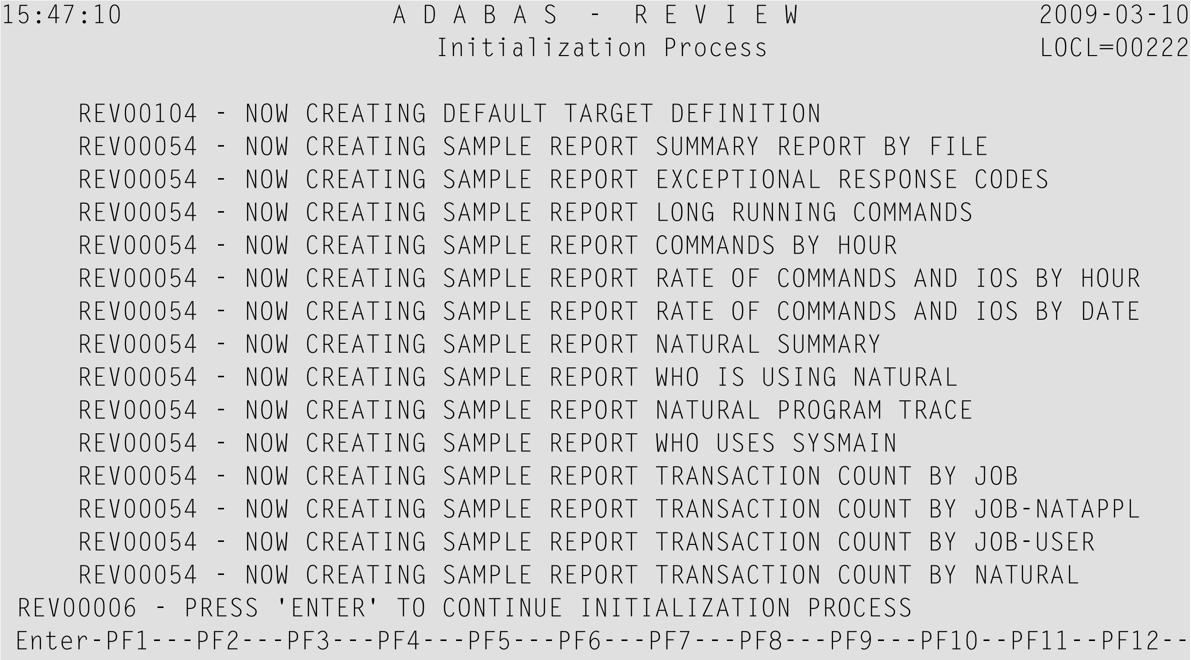
```
15:46:47                 A D A B A S  -  R E V I E W                2009-03-10
                                  Main Menu                         LOCL=00222

               +-------------------------------------------------+
               +-------------------------------------------------+
               |          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 any number in the SVC field (an Adabas SVC is not used in BS2000). Type the Adabas version number in the Version field, and press ENTER.

Based on the information you provide, Adabas Review creates the default target definition and displays a message. A *target* is a database monitored by Adabas Review. For more information about target objects, read *Displaying SVC Lists and Target Objects*, in the *Adabas Review Administration Guide*.

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



```
15:47:10                  A D A B A S  -  R E V I E W                  2009-03-10
                             Initialization Process                    LOCL=00222

     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

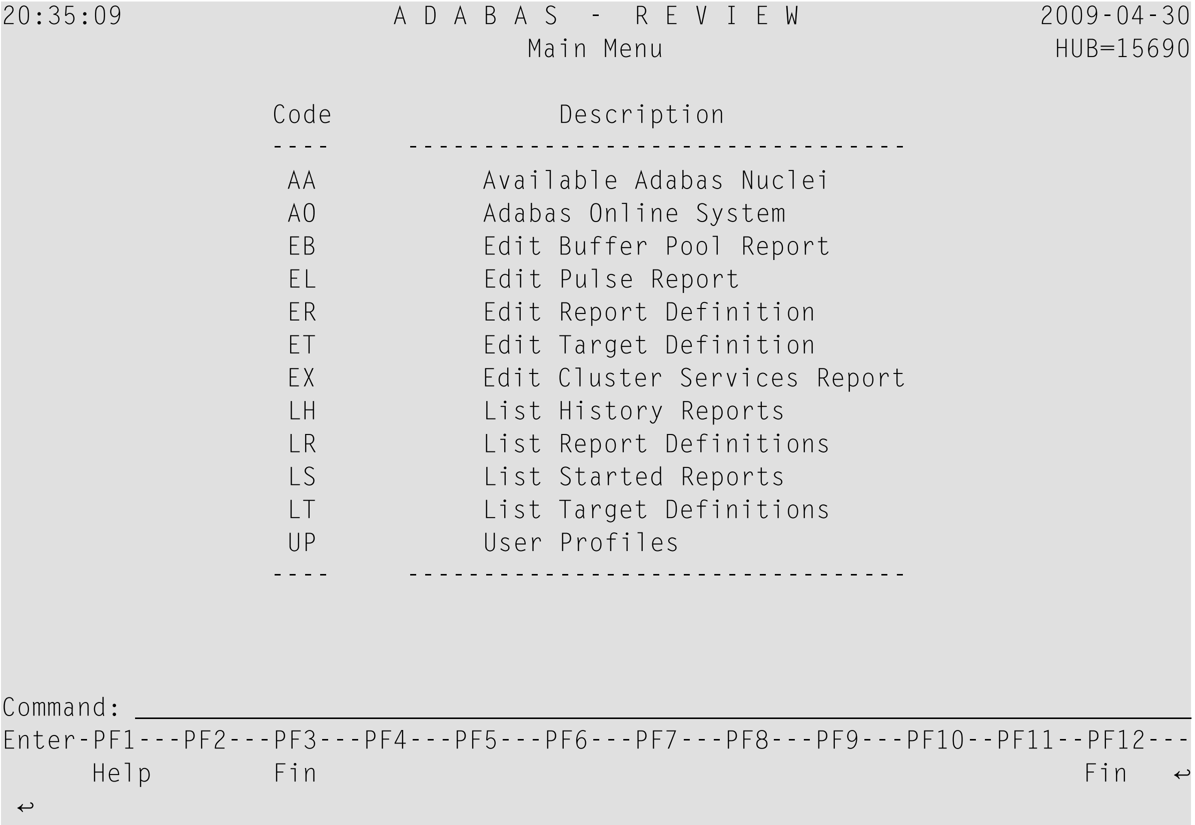
To access Adabas Review

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 Under TIAM/Batch or UTM

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:

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.

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.

6 Operations

- SYSOUT and a Subtask List Files 40
- Files Used by Adabas Review 41
- Editing the RVUEXI Parameter File 44
- Command Logging Considerations 45
- Summary Logging Considerations 47
- Modifying Configuration Parameters 49
- User Exit 5 (Adabas Review Hub Event Handler) 51
- Adabas Review Natural User Exits 53

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

SYSOUT and a Subtask List Files

Adabas Review spawns the following subtask jobs:

- In local mode, the Adabas Review database and in hub mode, the Adabas Review hub spawns a job where the Adabas Review subtask REVIEWB runs.
- When an autogenerated report is started, a job running REVAUTO is started.
- When a history report is started, a job running RAOSHIST is started.

Each of these jobs generates a SYSLST file with the following file name structure:

```
L.<L/O>.<task_number>,<subtask>,<timestamp>
```

where

L	is for SYSLST
O	is for SYSOUT
task_number	is the task number of the Adabas Review database or Adabas Review hub
subtask	is one of the following:
	REVIEWB# — for the Adabas Review subtask RAOSAUTO — for the autogenerated report subtask RAOSHIST — for the history report subtask
timestamp	is the STCK value when the subtask was started. In older Adabas versions (prior to 8.1.4), this value is in hexadecimal; in Adabas 8.1.4 and later versions, this value appears in the format: *yyyy-mm-dd.mmss.sss*.

The files provide important diagnostic information when errors occur within the system. They may, however, accumulate in number over time so that it becomes necessary to delete them.

To delete all accumulated Adabas Review subtask listings

- Use the commands

  ```
  /DELETE-FILE L.*REVIEWB#*
  /DELETE-FILE L.*RAOSAUTO*
  /DELETE-FILE L.*RAOSHIST*
  ```

Files Used by Adabas Review

RVLOGnn Command Logging Files

The RVLOG*nn* files are sequential command logging files. There may be more than two command logging files. If the files are given no secondary extent, data will be continued in the next RVLOG*nn* (*nn+1*) file. When the last command logging file has filled, command logging will start overwriting the data in RVLOG01. The P.GENERATE job can be amended to create more than two command logging files, as needed by your installation.

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 (files) for a particular report must be the same size.

Adabas Review command logging under BS2000 is an optional feature. BS2000 uses its own command logging files.

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

RVSUMnn Summary Logging Files

The RVSUM*nn* files are sequential summary logging files. There may be more than two summary logging files. If the files are given no secondary extent, data will be continued in the next RVSUM*nn* (*nn+1*) file. When the last summary logging file has filled, summary logging will start overwriting the data in RVSUM01. The P.GENERATE job can be amended to create more than two summary logging files, as needed by your installation.

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 (files) for a particular report must be the same size.

Adabas Review summary logging under BS2000 is an optional feature. BS2000 uses its own summary logging files.

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

RVUALT History File

RVUALT is an alternate sequential file used to save history data.

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.

The Adabas Review startup job contains a RVUALT job control statement. This statement identifies an alternate file to which historical data may be written.

In situations where the Adabas Review repository is unavailable, Adabas Review receives a response code 148.

In this case, Adabas Review writes the data to the file specified by the RVUALT job control statement, if it has been assigned in the job stream. The next time the Adabas Review hub is started, another subtask is started to copy the historical data to the Adabas Review repository.

Note: A separate RVUALT data set (file) must be allocated for each Adabas Review hub.

For BS2000 systems, allocate this file using P.GENERATE.

RVUAUT1 and RVUAUT2 Report Definition Data Sets (Files)

RVUAUT1 and RVUAUT2 are data sets (files) 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.

For BS2000 systems, generate these files using P.GENERATE.

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 (File) for GENCARD Command

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

For BS2000 systems, the command requires the user to supply a DDNAME, 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 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 (file) contains parameter control statements for creating user-defined fields. Parameters in this data set (file) define the length, type, and location of reporting fields to be determined by the user.

RVUPARM Dummy Data Set

Software AG recommends that you set RVUPARM as a dummy data set. 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 (file) if desired, and Adabas Review will access this data set (file) prior to accessing data sets (files) specified by RVUAUT1 and RVUAUT2.

For BS2000 systems, use the data set (file) *DUMMY.

Notes:

1. When RVUPARM is "dummied", the following message displays: **REV20164 - OPEN FAILED FOR RVUPARM**. When RVUPARM has been dummied, this message is normal and should be ignored.
2. The above error message can be avoided by creating a RVUPARM data set (file) 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 startup JCL (hub mode) or to the Adabas nucleus startup JCL (local mode).

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

RVUEXI is a parameter file that contains parameters which 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	1000

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	60

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	60

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 (files). Command log data sets (files) must be allocated for reports.
2. Add job control statements to the Adabas Review hub startup JCL (hub mode) or to the Adabas nucleus startup JCL (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 JCL.

   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 (files) 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 (files) 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 (files).

   Summary log data sets (files) must be allocated for reports.
2. Add job control statements to the Adabas Review hub startup JCL (hub mode) or to the Adabas nucleus startup JCL (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 JCL.

   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 (files) 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 (files) 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.

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 the CONFIGDB parameters

1. At the Natural NEXT prompt, type LOGON SYSREVDB and press ENTER.
2. Type the command EDIT CONFIGDB and press ENTER.
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 contains parameters that affect Adabas Review.

CONFIGDB is saved in the Natural library SYSREVDB.

Parameter	Possible Values	Default
CURSOR-POSITION	BOT | TOP	BOT

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=*value*	.

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	YES | NO	YES

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	YES | NO	NO

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	name	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 8 or greater.

Parameter	Possible Values	Default
OPEN-DBID	YES I NO	NO

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.

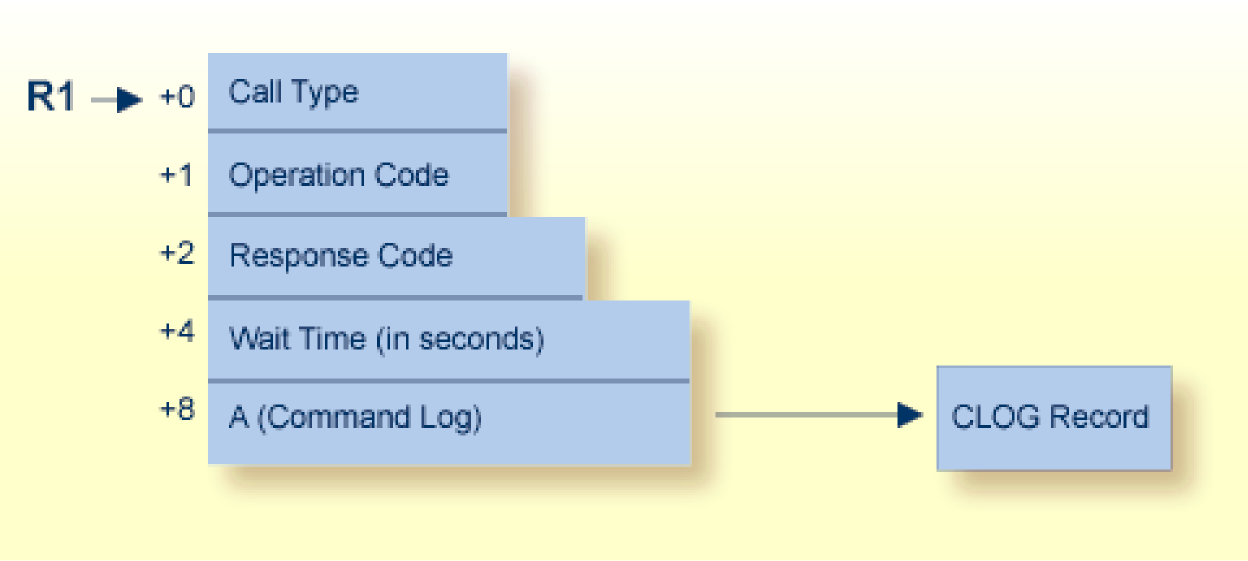
The exit is invoked with AMODE=31 and should return control in the same state.

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: ■ "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: ■ "W" -- wait for a specified time and then retry; ■ "R" -- retry logging operation immediately; or ■ "I" -- ignore the logging failure and continue without consequence.
2(R1)	Response code for logging errors. This response code is the same as the Adabas response code in the *Adabas Messages and Codes* 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 *not* 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 `SETUP`/`RETURN`).

7 ADARUN Parameters for Adabas Review in BS2000

- PROGRAM : Program to Run 56
- REVIEW : Adabas Review Control 57
- CMDQMODE : Command Queue Mode 58
- FORCE : Allow Nucleus Database ID or Review Hub Table Entry Overwrite 59
- NAB : Number of Attached Buffers 60
- NC : Number of Command Queue Elements 61
- REVFILTER : Review Record Filtering Control 62
- SUBMPSZ : GETMAIN Memory Pool for Subtasks 63

ADARUN performs the following functions:

- Loads the ADAIOR module, which performs all database I/O and other functions that depend on the operating system.
- 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 DDCARD 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 BS2000 environments.

PROGRAM : Program to Run

Parameter	Specify:	Possible Values	Default
PROGRAM	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 *Adabas Session Execution*, in *Adabas Operations* 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.

Specify:	To start:
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.
USER	a user program to be run using a non-reentrant Adabas batch/TSO link routine. For more information, read *Linking Applications to Adabas*, in *Adabas Operations* documentation.
utility-name	an Adabas utility Specify an Adabas utility for *utility-name*. For more information, refer to *Adabas Utilities* 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. **Note:** If an Adabas Review load library is not included in the startup JCL, the REVIEW parameter is automatically changed from LOCAL to NO.
dbid	Adabas Review is started in hub mode. The physical database ID that you specify for the hub identifies ■ 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). In hub mode, Adabas Review job control statements should be added to the Adabas Review hub startup JCL.

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
```

CMDQMODE : Command Queue Mode

This parameter applies to the BS2000 operating system only.

Parameter	Specify . . .	Possible Values	Default
CMDQMODE	whether to allocate the command queue memory pool below or above the 16-MB line.	BELOW | ABOVE	ABOVE (BELOW for Adabas versions prior to Version 8)

CMDQMODE specifies whether to allocate the BS2000 memory pool for the Adabas command queue below or above the 16-MB line.

Value	Meaning
BELOW	The default setting. Places the BS2000 memory pool for the Adabas command queue below the 16-MB line in one or more 64-kilobyte segments.
ABOVE	Places the BS2000 memory pool for the Adabas command queue above the 16-MB line in one or more 1-MB segments.

Example

The following example, places the Adabas command queue memory pool above the 16-MB line in 1-MB segments.

```
ADARUN PROG=ADANUC,CMDQMODE=ABOVE
```

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
```

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
```

SUBMPSZ : GETMAIN Memory Pool for Subtasks

Parameter	Specify . . .	Possible Values	Default
SUBMPSZ	the common memory pool size, in bytes, for subtask communication in products such as Adabas Review, Adabas Parallel Services, and Event Replicator for Adabas.	100000 - address-limit	1,024,000

For Adabas Review on BS2000 systems, the recommended value is 140,000,000 bytes.

Notes:

1. Setting this parameter for Adabas Review replaces an optional zap for increasing the subtask common memory.
2. This parameter must be set to the recommended value for the Adabas Review hub and the Adabas Review nuclei, irrespective of the value of the REVIEW parameter.

Example

The following example allows for four (4) megabytes of common memory pool storage for use in the commumication between the Adabas nucleus and the subtasks.

```
ADARUN PROG=ADANUC,SUBMPSZ=4096000
```

8 Operator Commands

- Entering Operator Commands 66
- Operator Command Overview 66

The commands in this section of the documentation are used to control the Adabas Review (ADAREV).

The operator commands 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.

The commands are listed alphabetically.

Entering Operator Commands

The ADAREV operator commands are entered the same way as other Adabas operator commands.

In BS2000 environments, enter each command at the operator console by addressing the Adabas nucleus with its task sequence number (TSN) in the following form:

```
/INTR  TSN,command
```

For testing purposes, the nucleus may be run as a dialog process. The nucleus may be interrupted by pressing the K2 key, after which the prompt "/" appears. Now an operator command can be sent to the nucleus in the following form:

```
/INTR  command
```

Note: In the dialog mode, the nucleus stops as long as the INTR message is not sent back. The resume statement /RESUME causes the nucleus to resume where it was interrupted when no operator command is issued.

Operator commands are processed by a STXIT routine.

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 a user completion code of 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:

```
REVH13 hub-id  001 CLIENT(S)  LAST-ACT RPT BUF PRTS WBUF LOG-RECORDS
REVH13 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

Terminate 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

Cancel Adabas session immediately. All command processing is immediately suspended. A pending AUTORESTART will be in effect which in turn will cause the AUTORESTART routine to be executed during the initialization of the next Adabas session.

Index

Symbols

/INTR operator command processing
  BS2000, 66

A

Adabas Native SQL
  BS2000 installation, 25
Adabas Review
  allocating storage for, 5
  parameter to set, 57
  starting under BS2000, 31
ADAEND operator command, 67-68
ADARUN parameters
  NAB to size the attached buffer, 5, 7
  NC to size the command queue, 5-6
  under BS2000, 55
attached buffer, 6
attached buffers
  number of
    parameter to specify, 60

B

BS2000
  entering operator commands, 66
  installation, 11
  installation of Adabas Native SQL support, 25
  installation of repository, 16
  installation under Adabas, 16-17
  installation under Natural, 14
  parameter for subtask GETMAIN memory pool, 63
  preparing for installation, 3, 11
  verifying the installation, 37
buffer pool
  attached
    space allocation, 60
BUFFER-SEGMENTS parameter, 7

C

CANCEL operator command, 67-68
CMDQMODE
  ADARUN parameter, 58
command logging
  considerations, 45
  setting up, 46
  user exit, 46
command logging files, 41
command queue, 6
  parameter to specify location of memory pool, 58
command queue element
  maximum number of
    parameter to specify, 61

D

DCQ operator command, 67
DECIMAL-CHAR parameter
  CONFIGDB file, 49
display
  list of queued requests, 67
  queued requests, 68
DNC operator command, 68

E

entering operator commands
  under BS2000, 66
environments
  supported, v, 4
examples
  /INTR for operator commands, 66

F

files
  command logging, 41
  summary logging, 41
FORCE
  ADARUN parameter, 59

H

hub server
  attached buffer storage requirements, 6
  command queue storage requirements, 6
  storage requirements, 5

I

ID Table
  parameter to
    allow nucleus to overwrite existing entry, 59
initialization, 33
installation

Adabas Review repository, 16
BS2000, 11
BS2000 in hub mode, 10
BS2000 in local mode, 10
BS2000 support for Adabas Native SQL, 25
BS2000 under Adabas, 16-17
BS2000 under Natural, 14
overview, 10
preparing for BS2000, 3
tape description, 9

L

local mode
defined, v
LOGUEXIT sample user exit, 47-48

M

MAXSTORE
report parameter, 8

N

NAB
ADARUN parameter, 60
NATPARM
parameter settings, 14
Natural
NTFILE definition, 15
required parameter settings, 14
user exits, 53
NC
ADARUN parameter, 61
nucleus operator commands, 66

O

operator commands, 66
entering under BS2000, 66

P

P-UEXITn, 53
PROGRAM
ADARUN parameter, 56

R

RAOHIST, 40
RAOSAUTO, 40
RAOSHIST, 40
reports
MAXSTORE parameter, 8
storage for control blocks, 8
storage for data collection areas, 8
storage requirements, 5, 8
total storage limit (MAXSTORE), 8
REVAUTO, 40
REVFILTER
ADARUN parameter, 62
REVIEW
ADARUN parameter, 57
REVIEW-BUFFER
storage requirements, 5, 7
REVIEWB, 40
REVIEWB#, 40
RVLOGnn command logging files, 41
RVSUMnn summary logging files, 41
RVUFLD
parameter file, 43

S

STARTCLIENT operator command, 68
starting Adabas Review, 31
STOPCLIENT operator command, 68
Storage
allocating above the 16MB line, 5
storage
MAXSTORE report parameter, 8
requirements, 5
STXIT operator command processing, 66
SUBMPSZ
ADARUN parameter, 63
subtask
Adabas Review, 40
autogenerated report, 40
deleting list files, 40
history report, 40
jobs, 40
list files, 40
summary logging
considerations, 47
setting up, 47
user exit, 48
summary logging files, 41
SYSLST
file name structure, 40
SYSOUT
file name structure, 40
System Maintenance Aid (SMA), 4

T

target
defining the default, 35
definition, 35
TP monitors
supported by BS2000, 4

U

UEX5, 51
user exits
exit 5, 51
hub event handler, 51
Natural, 53
users
storage requirements for online use, 5, 9

W

work areas
storage requirements, 5, 9