

Adabas Review

Concepts and Facilities

Version 4.4.1

June 2008

This document applies to Adabas Review Version 4.4.1 and to all subsequent releases.

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

Copyright © Software AG 2008. All rights reserved.

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

Table of Contents

1 Concepts and Facilities	1
2 Introduction	3
Collecting Data in Local Mode	4
Collecting Data in Hub Mode	5
Collecting History Data	12
Processing Data in Batch Mode	12
Reports	13
Autostarted Reports	14
Repositories	15
User Profile System	15
Storage Requirements	16
ADARUN Parameters	20
3 About Review Reports	21
Overview	22
Interactive and Batch Reports	22
Report Commands	23
Components of a Report Definition	23
Report Display Program	26
List Report Functions	27
Record Filtering	27
4 Summary of Features	31
Index	33

1 Concepts and Facilities

•	<i>Introduction</i>
•	<i>About Review Reports</i>
•	<i>Summary of Features</i>

2 Introduction

■ Collecting Data in Local Mode	4
■ Collecting Data in Hub Mode	5
■ Collecting History Data	12
■ Processing Data in Batch Mode	12
■ Reports	13
■ Autostarted Reports	14
■ Repositories	15
■ User Profile System	15
■ Storage Requirements	16
■ ADARUN Parameters	20

Adabas Review provides 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.

Adabas Review runs in

- local mode in the Adabas address space. Under z/VM, Adabas Review runs in local mode only.
- hub mode as a server in its own address space with a client interface in the Adabas address space.
- batch mode as a batch job that processes sequential Adabas command log data sets

See the *Release Notes* for this version of Adabas Review for a matrix of supported Adabas versions and other requirements.

This chapter covers the following topics:

Collecting Data in Local Mode

In local mode, the Adabas Review processor is installed as an extension to ADALOG.

The data collection process is partly accomplished by the Adabas Review processor. The data is collected by providing code (RAOSLOCL) that runs as an extension to ADALOG. Adabas calls this module and passes information about resource usage for each command processed by the Adabas nucleus.

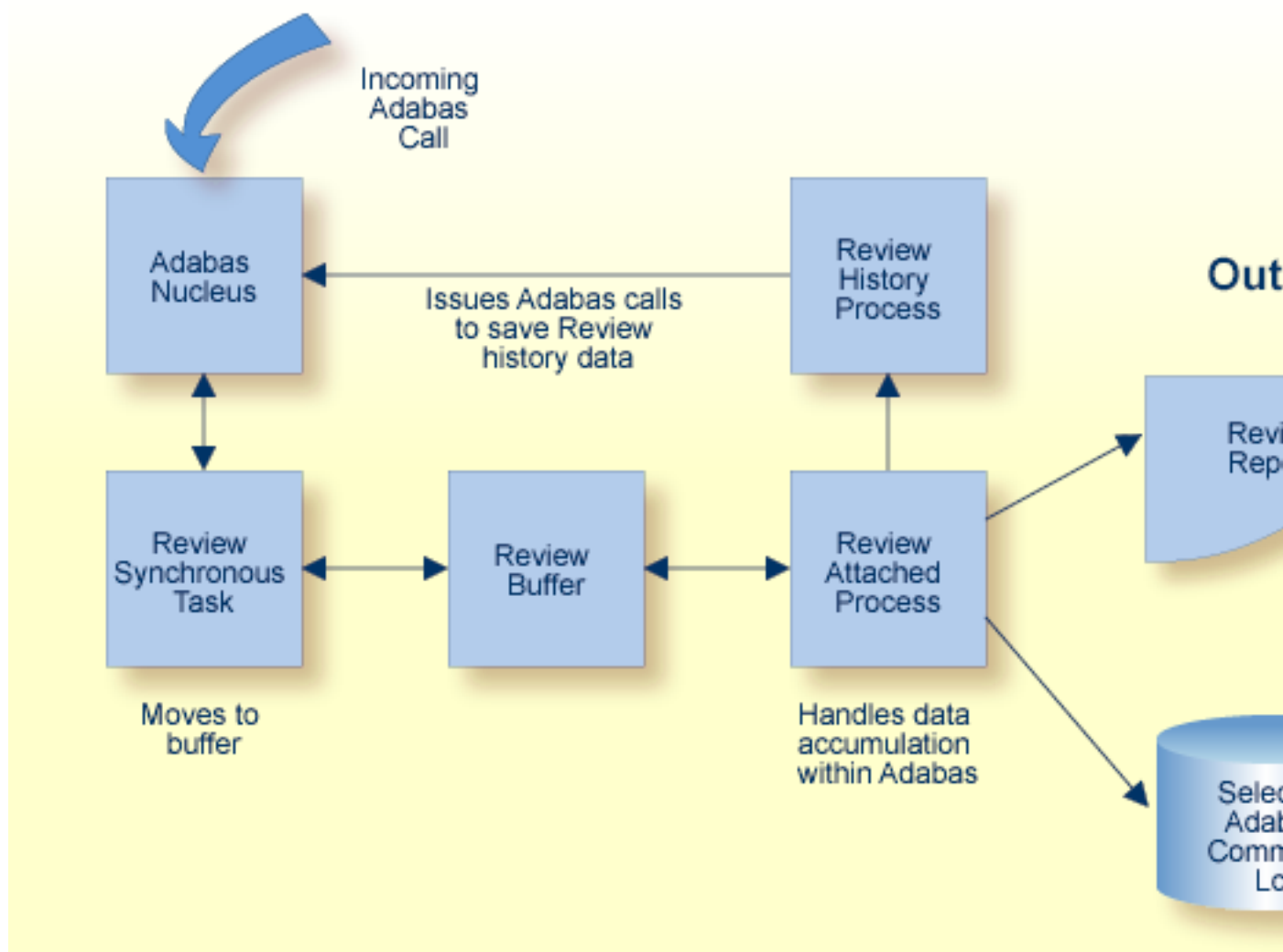
This module, in conjunction with the Adabas Review processor and an intermediate Review buffer, accumulates and tabulates the Adabas information based on various user-defined data requirements.

The Adabas Review data may be

- displayed in an online environment from the Adabas Review user interface;
- saved automatically in the Adabas Review repository;
- printed automatically when the Adabas nucleus terminates; or
- downloaded directly to a personal computer (PC) using Entire Connection.

Collecting Data in Local Mode

The following graphic shows the Adabas Review data collection process for local mode.



Collecting Data in Hub Mode

In hub mode, Adabas Review uses a client/server approach to collecting data:

- an interface (the client) resides on each Adabas nucleus; and
- the hub (the server) resides in its own address space, partition, or region.

The interface uses the existing Adabas interregion communication process: ADALINK, TRANSPORT, ADASVC (ADARER), and ADAMPM. This process is consistent across the targeted platforms

for Adabas Review. If systems are networked correctly, hub mode supports a multiple platform, multiple operating system, Adabas database environment.



Note: TRANS-PORT is available only in z/OS environments.

The Server Hub

The Adabas Review hub is a centralized data collector and reporting interface that combines proven components of Adabas and Review.

It handles the data consolidation and reporting functions for monitoring an Adabas database, including usage information related to applications, commands, command response time, I/O activity, buffer efficiency, and others.

The interactive reporting facility allows you to pinpoint problems quickly, providing detailed and summary data about Adabas activities. Specific information about each database is also available.

The centralized collection server has several advantages:

- A single hub collects information from multiple Adabas nuclei, Adabas Parallel Services clusters, or Adabas Cluster Services (support for IBM's parallel sysplex environment) clusters.
- Because a single hub can support multiple Adabas nuclei, the number of Adabas Review nuclei required to support an enterprise-wide distribution of Adabas nuclei is reduced. This minimizes resource requirements and increases performance.
- Isolating the Review subtask from the Adabas nucleus enhances the performance of the Adabas main task and minimizes the impact of future Adabas releases on the functioning of Adabas Review.

The hub comprises

- ADAREV, a logic module that manages and supervises the incoming Review data calls and requests;
- REVHUB, a module to establish and maintain the environmentals for Adabas Review; and
- the Review DB nucleus and subsystems including RAOSAUTO, the autostarted report parameter generation routine, and RAOSHIST, the historical data population routine.

The Client Interface

The Adabas Review interface constructs and then transmits the Review data from the Adabas nucleus to the Adabas Review hub. An Adabas Review interface is integrated with each Adabas nucleus that is monitored.

The interface comprises

- ADALOG, the Adabas command logging module;
- ADACLX, the Adabas command log extension module that is responsible for acquiring additional information not present in the Adabas command log record; and
- ADARVU, which handles the environment conditions for ADACLX and the Adabas API requirements for transmitting the Review data to the Adabas Review hub.

Interface Calls

To maximize performance, the ADARVU module issues an "optimistic" call from an ADABAS nucleus to the Adabas Review hub without waiting for a completion or "post" from the hub; ADARVU assumes that the Review data was successfully passed to the hub.

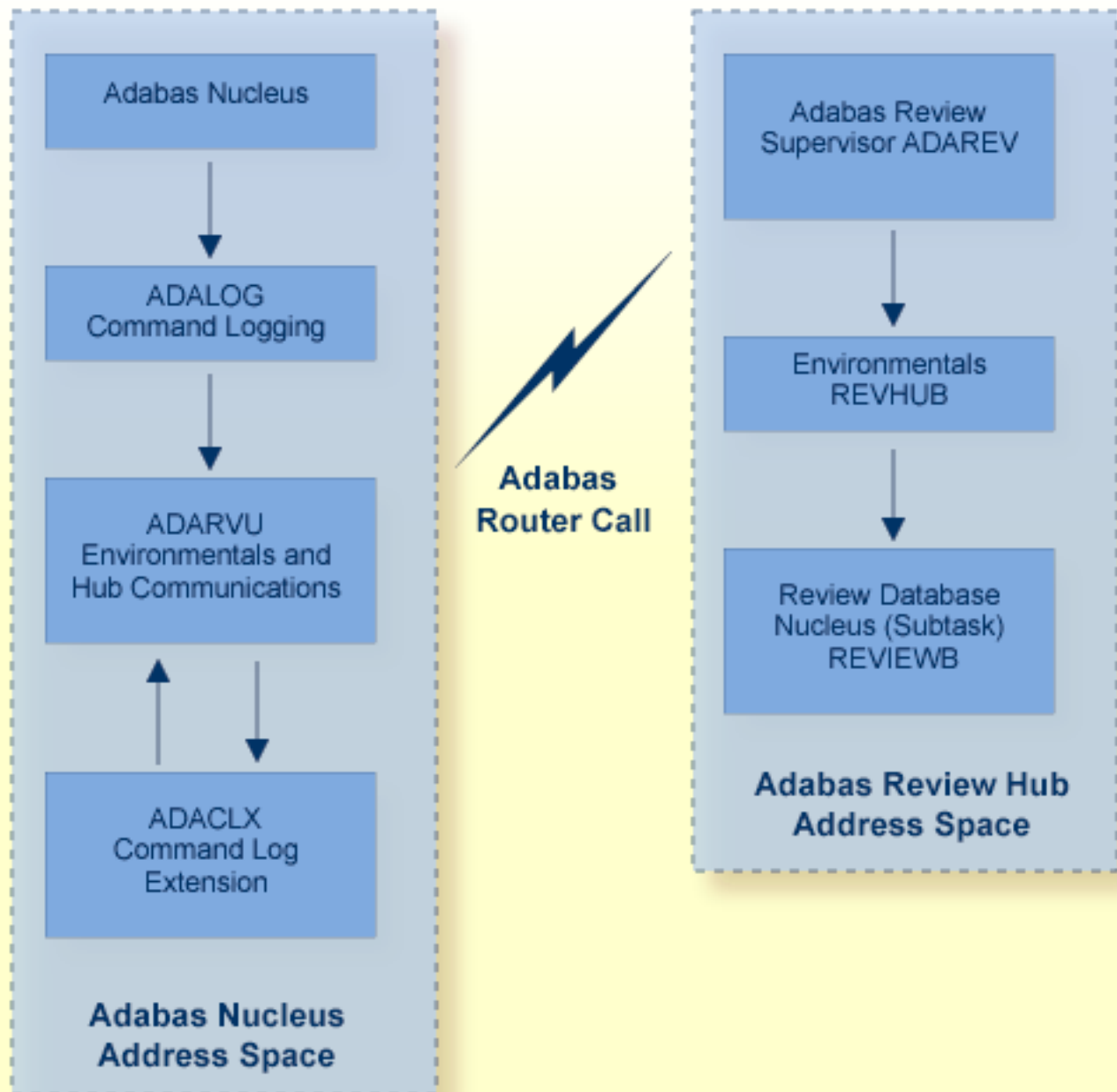
However, ADARVU does perform an initialization step to ensure that the hub is active prior to any command processing by the Adabas nucleus. If the hub is not active, ADARVU informs you using WTOs and/or a user exit. If a user exit is used, you are given the option to wait for the hub to be activated, or continue initialization and call the hub only when it is active.

On the hub side of the call, the elimination of the cross-memory "post" call enhances performance by reducing the overhead of active communication with the Adabas clients. This allows the hub to remain a passive data collector.

Example Client/Server Environment

The following graphic shows the major components of the Adabas Review interface (Adabas nucleus address space) and hub (Adabas Review hub address space) in a client/server architecture

.



Command Log Processing

The data collection process is partly accomplished by the hub (server) component REVIEWB, the Adabas Review command log processing routine, which runs as an Adabas Review *subtask*; that is, a unit of work that the operating system treats as separately detachable.

At initialization, REVIEWB reads any autostarted report definitions the user has defined and collects data according to the reports' criteria. REVIEWB also processes requests to start, view, and purge reports from the Adabas Review online system.

In hub mode, Adabas responds to requests and calls the interface module ADARVU from ADALOG (Adabas's command logging module) if REVIEW=dbid is specified in the Adabas initialization parameters. Adabas passes to ADARVU information about resource usage for each command processed by the Adabas nucleus.

Adabas Review link routine exits are used to pass TP system and Natural information from the user's address space (origin of the Adabas call) to the Adabas address space and, using an extension of the Adabas user buffer, on to ADARVU.

ADARVU queues Adabas command log records received from ADALOG to REVIEWB through an intermediate REVIEW-BUFFER in the hub (server) address space. Note that command log records are examined prior to queuing to determine whether record filtering is active. Command log records are sent to the hub only when record filtering is active.

See [Record Filtering](#) for more information.

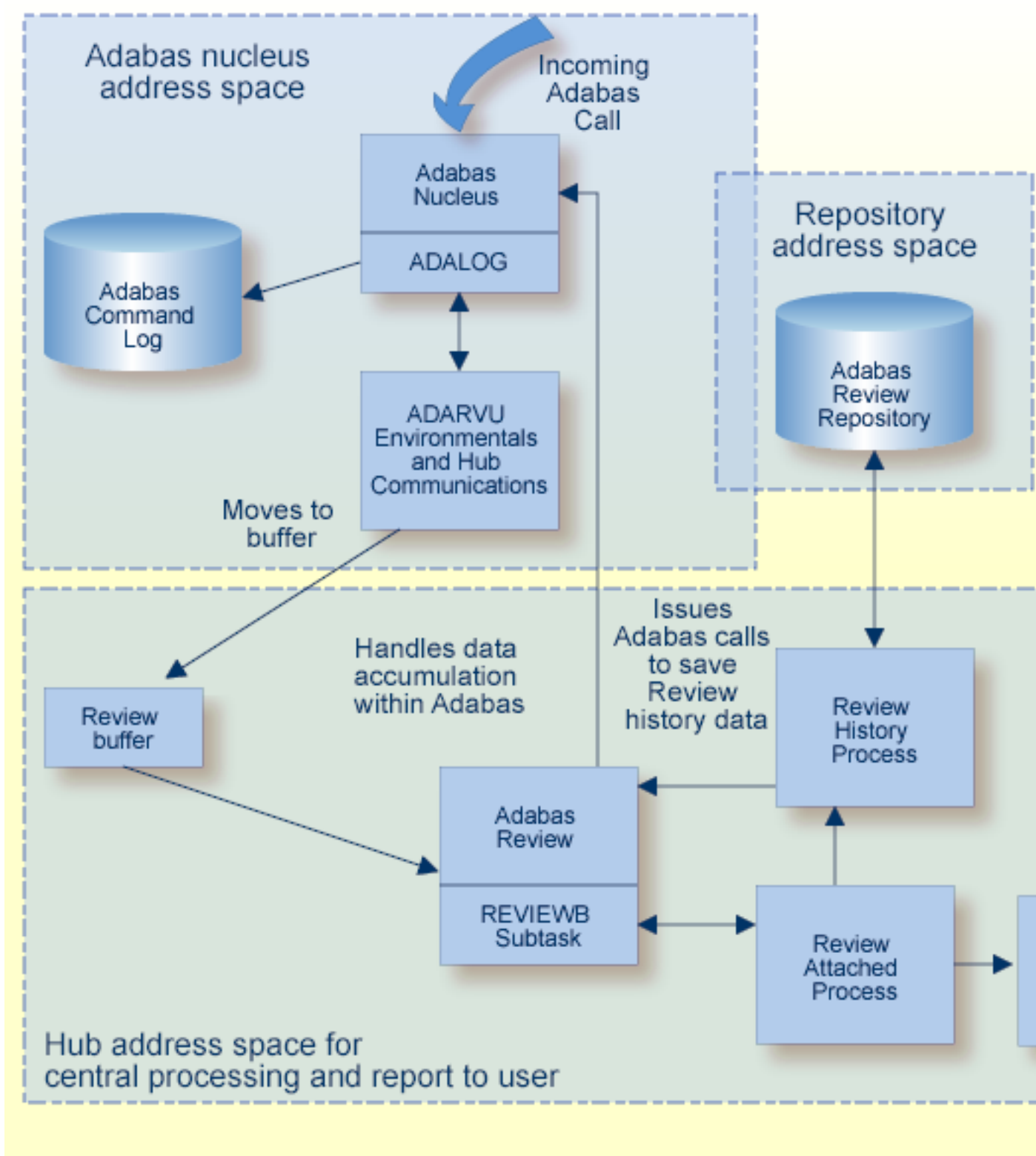
REVIEWB processes the records, accumulating and tabulating various data according to the criteria specified in any user-defined reports that are active.

The resulting nucleus statistics may be

- displayed in an online environment from the Adabas Review user interface;
- saved automatically in an Adabas file called the Adabas Review repository;
- printed automatically when the Adabas nucleus terminates; or
- downloaded directly to a personal computer (PC) using Entire Connection.

Example Hub Mode Data Collection Process

The following graphic shows the Adabas Review data collection process for hub mode. When monitoring multiple databases, Adabas Review allows you to switch from one database to another and provide reports for each.



Collecting History Data

History data collection is controlled by RAOSHIST, the Adabas Review historical data population routine.

z/OS, z/VSE, and BS2000

- RAOSHIST executes as a subtask of Adabas Review. At initialization, RAOSHIST reads the RVUALT data set to determine if there are any historical records from the previous Adabas Review session that should be written to the Adabas Review repository. History records are written to RVUALT if the database on which the Adabas Review repository resides is unavailable during Adabas Review termination.
- During normal execution of Adabas Review, REVIEWB is responsible for adding history records generated by detail history reports and by summary history reports that have a history interval.

z/VM

RAOSHIST is executed by an EXEC or a job, and adds records read from the RVUALT file to the Adabas Review repository file. All history records are written to the RVUALT file in z/VM.

Processing Data in Batch Mode

As a batch job, the Adabas Review command log processing routine REVIEWB processes Adabas command log records from a sequential data set. If you use Adabas dual command logging, you must first use the Adabas utility function ADARES CLCOPY to generate a sequential command log data set suitable for input into Adabas Review.

When Adabas Review executes as a batch job, input report parameters that define the data collection criteria selected by the user are read from statements in the RVUPARM data set or the RVUAUT1/RVUAUT2 data sets. These statements can be generated using the GENCARD statement.

The storage allocated for reports is exactly the same as that for Adabas Review executing in interactive (online) mode. However, since REVIEWB is reading the command log records directly from a sequential file, no REVIEW-BUFFER is allocated.

Reports

Online or as a stand-alone batch job, Adabas Review processes Adabas command log records and generates reports according to user-defined reporting criteria. The flexible reporting structure of Adabas Review allows you to view the same data in many different ways.

- [Report Definition](#)
- [Natural Display Program](#)
- [Started Reports](#)

Report Definition

Adabas Review uses a set of instructions called a **report definition** to specify the types of data to be collected. Prepared report definitions supplied with Adabas Review may be modified and custom reports may be created.

Report definitions can be created or modified using menu-driven Natural programs. Report options and processing rules allow you to specify the conditions under which the data is to be captured. Report definitions are kept in the Adabas Review repository.

Natural Display Program

When a report definition is saved, Adabas Review generates a Natural display program that is executed when you wish to view the collected data. This display program is executed

- to view data currently being collected by the interactive Adabas Review processor.
- to retrieve and display or download historical data that has been saved and stored in the Adabas Review repository.



Note: No display program is created for buffer pool reports.

The Natural display program generates normal Natural `FIND` and `READ` statements against a Review DDM to access the data being collected by Adabas Review. By default, the Natural program displays the data at the Adabas Review user's terminal. Options exist, however, to download the data directly to a personal computer (PC).

Started Reports

Once the report definitions are edited and saved, the reports can be started. Starting a report tells the Adabas Review data collection process to start accumulating data based on the report definition parameters.

Adabas Review users can display buffer pool information, display active databases, and access the Adabas Online System, if it is available. Adabas Review administrators are also allowed to define and display target objects.

Autostarted Reports

Adabas Review reports can be set to start automatically whenever Review initializes.

Then RAOSAUTO, the Adabas Review autostarted report parameter generation routine, generates the report definition control statements and writes them to one of two parameter files, RVUAUT1 or RVUAUT2, alternating between them by writing to the older file.

When Adabas starts, the files are read by Adabas Review using the RVUAUT1 and RVUAUT2 statements in the job stream.



Note: Under z/OS, the installation procedure defines the statements RVUAUT1 and RVUAUT2 so that they point to members of a PDS. To avoid constant compression of these data sets, the statements may point to sequential data sets.

RAOSAUTO automatically regenerates the control statements for all autostarted reports when you make changes to an autostarted report, delete an autostarted report, or modify the target definition for the database being monitored by the reports.

In exceptional circumstances (e.g., the source library becomes too full and requires compressing), you can force regeneration of the control statements for all autostarted reports by either issuing the GENAUTO command or entering the parameters manually using batch parameter statements.

Additionally, when you issue the GENCARD command, RAOSAUTO generates report parameter cards for user-specified reports and directs them to a user-specified output file.

In z/OS, z/VSE, and BS2000, RAOSAUTO executes as a subtask of Adabas Review and is only active when

- an autostarted report definition is saved in the online system;
- GENAUTO (or GENCARD) is executed in the online system; or
- a database target definition is saved in the online system.

In *z/VM*, *RAO**S**AUTO* executes as an *EXEC*. The *GENAUTO* and *GENCARD* commands are not available in *z/VM*.

Repositories

The Adabas Review repository is an Adabas file used for storing report definitions, historical data, and target definitions. In hub mode, it cannot run in the hub address space.

Depending on the configuration at your site, more than one Adabas Review repository may be associated with your system. For example, if your site is running Adabas Review against more than one database, you may choose to have an Adabas Review repository for each database.

The Review command *SETFILE* (or *SET*) may be used to access different Adabas Review repositories and the reports stored on them.

User Profile System

Adabas Review administrators use the user profile system to generate profiles that define access rules for Review users. Access rules specify the systems or the functions within systems that a particular user is allowed to use.

User profiles may be created for new users, changed for existing users, and purged when no longer required.

A user profile is not required for each user. Review provides a default profile to allow access for users who do not have a profile defined.

When a user logs on, Review searches for the user's profile. If one is not found, the default profile is used.

If the default profile is customized so that the access rules meet the needs of the majority of Review users, the need for individual user profiles can be eliminated.

If a user has access needs that are different from the majority, a user profile can be created to accommodate those needs. Such a profile is generated by customizing a copy of the default profile.

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.

In z/OS , Adabas Review allocates all storage from z/OS subpool 5. This allows you to accurately determine the exact amount of storage Adabas Review is using with a z/OS monitoring package.

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). See the *Adabas Operations Documentation* for more information.

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.

The Command Queue and TRANS-PORT

When using TRANS-PORT, most communication is handled via a fast buffer mechanism using a fixed buffer length. In such cases, the command queue is not used.

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, i.e., whether control buffers are required, or 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.

The Attached Buffer and TRANS-PORT

When using TRANS-PORT, most communication is handled via a fast buffer mechanism using a fixed buffer length. The attached buffer area is not used.

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 128 bytes. The default value for `BUFFER-SEGMENTS` is as follows:

Operating System	Default Value
z/OS	2000 (250 kilobytes)
z/VSE	700 (87 kilobytes)
z/VM, or BS2000	400 (50 kilobytes)

For z/OS, z/VSE, and BS2000

For z/OS, z/VSE (version 1.3 and above), 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.

For z/VM

In z/VM systems, the fact that REVIEWB does not execute as a subtask of Adabas Review but is called synchronously only when the REVIEW-BUFFER is full has the following consequences:

- A **larger** REVIEW-BUFFER means that REVIEWB is called less often but with more command log records to process. Since Adabas is unable to process incoming calls until REVIEWB has processed the contents of the REVIEW-BUFFER, a considerable delay may be experienced when REVIEWB is called.
- A **smaller** REVIEW-BUFFER means that REVIEWB is called more often but with fewer command log records to process so that the processing delay is less for each processing cycle.



Note: The Adabas RVUEXI parameter `CMS-FULLSYNCH` may be used to force REVIEWB to process each command log instead of waiting for the REVIEW-BUFFER to become 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 $\frac{1}{2}$ 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.

ADARUN Parameters

Adabas ADARUN initialization parameters define and start the Adabas operating environment. Different ADARUN parameters are useful to the Adabas nucleus and the Adabas Review hub in different operating environments. For more information about the ADARUN parameters specific for Adabas Review in a given environment, select the appropriate link below:

- *ADARUN Parameters for Adabas Review in z/OS*
- *ADARUN Parameters for Adabas Review in z/VSE*
- *ADARUN Parameters for Adabas Review in BS2000*

3

About Review Reports

■ Overview	22
■ Interactive and Batch Reports	22
■ Report Commands	23
■ Components of a Report Definition	23
■ Report Display Program	26
■ List Report Functions	27
■ Record Filtering	27

This chapter describes Adabas Review report types and components, and discusses the display programs used for report output:

Overview

This section provides an overview of Adabas Review reporting features.

What Does Adabas Review Report?

Adabas Review retrieves performance information about Adabas by collecting data in the form of reports.

Reports capture command log information and provide information about both online and batch transactions. Reports collect data about multiple teleprocessing environments and batch programs that are all using the same database.

What is an Adabas Review Report?

An Adabas Review report has two parts: a definition and a display program.

- A *report definition* is a set of parameters that specify the data to be captured and a set of processing rules that identify the conditions under which the data is to be captured.
- A *report display program* is a Natural program that specifies the appearance of the report output, whether viewed online or downloaded to a personal computer (PC).

How are Adabas Review Reports created?

The Edit Report (ER) function is used to create Adabas Review report definitions. In addition, Software AG delivers prepared report definitions called *supplied reports* with Adabas Review. Supplied reports may be modified to meet the requirements at a particular site. Detailed information about supplied reports is provided in the *Adabas Review User's Guide*.

Interactive and Batch Reports

Reports can be run in either interactive or batch mode:

- In *interactive mode*, Adabas passes information to Adabas Review about resource usage for each command processed by the Adabas nucleus. Online reports are started with the `START` command.
- In *batch mode*, Adabas Review runs as a batch job that processes sequential Adabas or Adabas Review command log data sets. Batch parameter statements are generated for the report and submitted in the job stream.

Report Commands

Adabas Review provides commands to

- modify report definitions;
- modify display programs;
- control report data collection; and
- direct report output.

Components of a Report Definition

An Adabas Review report definition comprises database fields to be monitored, processing rules, and option parameters, all either entered on screens or selected from lists:

- **database fields**
from which a report extracts data are listed in the order in which they are to be displayed.
- **processing rules**
identify the conditions under which the data is to be captured by describing how field values are to be extracted for the report.
- **report option parameters**
specify whether Adabas Review will perform physical command logging, and whether data accumulated by the report will be written to a history file. History data may be used when analyzing database performance trends.

Database Fields

The selection of database fields defines the subject of the Adabas Review report.

A field identifies a type of information to be collected by the report; for example, information about an Adabas command, the Natural program issuing the command, the Adabas nucleus, the teleprocessing monitor, or the operating system being used. The PF11 (Flds) option is used to display a list of all available Adabas Review report fields.

Up to 20 fields may be used in a particular report. However, the number of fields that can be displayed is limited to the total line size (250 bytes) that can be displayed when a report is viewed online.

Fields to be displayed in a report are entered on the Edit Report screen of the Edit Report (ER) function as shown in the following example:

20:35:15A D A B A S - R E V I E WYYYY-MM-DD

Edit ReportLOCL=00009

Report Name: _____

DBID to Monitor: _____

Field	Order	Sum	Min	Max	Avg	Pct	Rate	Round
_____	_____	-	-	-	-	-	-	_____
_____	_____	-	-	-	-	-	-	_____
_____	_____	-	-	-	-	-	-	_____
_____	_____	-	-	-	-	-	-	_____
_____	_____	-	-	-	-	-	-	_____
_____	_____	-	-	-	-	-	-	_____
_____	_____	-	-	-	-	-	-	_____
_____	_____	-	-	-	-	-	-	_____
_____	_____	-	-	-	-	-	-	_____

Page 1

Command: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---

Help Optns ExitSave Start - +Rules Flds Menu

Detailed information about individual fields is provided in the *Adabas Review User's Guide*.

Report Processing Rules

Report processing rules are used to set conditions on or limit the data collected by the report.

Processing rules are entered on the Report Processing Rules screen of the Edit Report (ER) function as shown in the following example:

20:24:56A D A B A S - R E V I E WYYYY-MM-DD

Report Processing RulesLOCL=00009

REPORT ONE

Field	Op	Value	And/Or
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____

----- Page 1 -----

Command: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---

Help Optns Exit Save Start - + Flds Menu

An Adabas Review report processing rule is specified by a field, a relational operator (equal to, less than, greater than, not equal to), and a specific value or range of values.

If there are multiple processing rules, the appropriate logical operator (AND or OR) must also be specified.

Fields used in processing rules are not displayed in the report results unless they are listed on the Edit Report screen.

Report Option Parameters

Values for report option parameters can be entered on the Report Options screen of the Edit Report (ER) function as shown in the following example:

```

19:30:50          ***** R E V I E W *****          YYYY-MM-DD
                  Report Options For:                LOCL=00222

+----- Report ----- Logging ----- History -----+
| Detail/Sum ..... S   Log ..... N   History ..... N |
| AutoStart ..... N   File ..... RVLOG   History Int ... ____ |
| Break ..... Y   Num of Logs ..... _2   HISTORY DBID ... __223 |
| Wrapping ..... N   Log Size ..... 99999   HISTORY FNR .... __22 |
| Print ..... Y   User Exit ... ____   History SVC ..... 237 |
| Rstrt/Intrvl Y ____ |
| Max Restarts:  999999 |
| Max K ..... _16   Buffers to Log |
| ADALimit ..... _1 |
| Display By .. SORTED   CB: Y   FB: Y   SB: Y |
| Entries ..... 999999   RB: Y   VB: Y   IB: Y |
| Limit ..... 99999999   IO: Y   EX1: N   EX2: N |
| Page/Line... _55 / 133 |
|
+-----+

Report Exits
-----
Detail Exit.. ____
Summary Exit: ____
Cmd .... CL Sum .... N

Command: _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Help Exit Acpt Dflts Menu

```

Adabas Review is delivered with report options defaults:

- You can modify these defaults for all subsequent reports.
- You can modify the default values for a particular report, either new or existing.

Detailed information about using the Report Options screen is provided in the *Adabas Review User documentation*.

Report User Exits

Adabas Review provides two report user exits: one for detail reports and one for summary reports.

- A *detail report user exit* is driven when a command log record is selected for the report. Only records that pass the processing rules are provided to the user exit. This exit may be used to create SMF records, accounting records, or for any other purpose.
- A *summary report user exit* is driven when a specified Adabas command is selected for the report; and/or a report is summarized. You may control the conditions that trigger the exit.

When creating a report, the user exit is specified on the Report Options screen or in the batch REPORT statement. The actual report user exit must be provided in an executable library accessible to Adabas Review.

Report Display Program

Once data has been accumulated for a report, the results can be

- saved to a file;
- viewed online;
- downloaded to a personal computer (PC) using Entire Connection; or
- printed when the Adabas nucleus terminates.

Once a report has been defined, the SAVE command is issued and the report definition is written to the Adabas Review repository.

A unique Natural program, the Adabas Review report display program, is generated automatically whenever a report is created or modified using the Edit Report (ER) function. This program determines how the report data will be viewed online or downloaded to a PC. It controls the appearance of the report output for an active database or for historical data.

You may use the Review-generated display program as it is created, or you may modify it for your individual needs.

The display program is executed when the `VIEW` command is issued to display the report results online.



Note: The output of reports that print at database termination is not controlled by a display program. At database termination, a `SYSOUT` from Adabas is written by `REVIEWB`. The format of this output cannot be modified by the Adabas Review user.

List Report Functions

Adabas Review provides three list functions for reports, which are accessible from the Adabas Review (DB system) menu:

Code	List . . .	This function lists . . .
LR	Report	existing Adabas Review reports by name. Reports may be edited, started, and purged using the commands in this function.
LH	History Reports	reports that have accumulated history data in the Adabas Review repository.
LS	Started Reports	reports that are currently accumulating data.

The lists displayed by the `LH` and `LS` functions are subsets of the list displayed by the `LR` function.

From the resulting lists, certain commands can be issued against a particular report that allow you to maintain report definitions, manage started reports, and manage the output they produce.

The commands are entered on the selection line preceding the name of the report.

Record Filtering

Review record filtering can decrease the number of command log (CLOG) records passed from the database to Review. When filtering is active, only those records needed for report processing will be passed to Review. Record filtering operation is based upon the reports currently active and cannot be directly controlled. However, the Adabas `ADARUN` parameter `REVFILTER` may be used to control whether the database allows filtering. See the appropriate installation documentation for a description of the `REVFILTER` parameter.

Record filtering is limited to and dependent upon select Review fields (filter fields). Reports with rules based solely on filter fields are eligible to be filtering reports. Any other report is a non-filtering report. Record filtering will be active only when all reports running are filtering reports. That is, record filtering will stop if any non-filter report is started. An example of a filtering report would be the *Exceptional Response Code* report, which has a single rule for the `RSP` field. If this is the only report running, only those command logs with exceptional response codes will be sent to Review.



Note: Record filtering is intended for those installations that ordinarily run very few reports (e.g., *Exceptional Response Codes*) at a time. A large number of active filtering reports will negate the performance benefits of record filtering. If your installation runs many rule-based reports, running with the ADARUN REVFILTER=NO setting is recommended. Some analysis of active reports may be necessary to fully realize filtering performance benefits.

Record Filtering Restrictions

The following restrictions apply for Review record filtering:

- For filtering to be activated, all currently active reports must have processing rules based solely on filtering fields.
- Wildcard values are not permitted. Alphanumeric values containing an asterisk (*) will result in filtering being turned off.
- EXCLUDE statements are not supported. An EXCLUDE statement will result in filtering being turned off.
- Special Review reports are non-filtering reports.
- CLOGLAYOUT=8 is required for record filtering.
- Filtering will not be active when running with LOGCLEX=YES.
- In Hub mode, any active report using the artificial target ID 11111 will result in filtering being turned off.

Review Fields Eligible for Record Filtering

The following Review fields can be used in rule processing in order to activate Review Record Filtering. The use of any other field in rules for active reports will cause filtering to be stopped.

See the Field Reference for detailed information about particular fields.

DBID

NUCID

CMDTYPE

JOBNAME

ASSOIO

DATAIO

WORKIO

CMD

RSP

CID

ACBDBID

FILE

ISN

ISNQ

ACBUSER

TPUSERID

TRANSID

STEPNAME

NATAPPL

NATPROG

NATUID

NATSTMT

4 Summary of Features

The major features of Adabas Review include the following:

- Supports *multiple Adabas nuclei clustered under a single database ID*, either in a single operating system image (Adabas Parallel Services) or under z/OS and IBM parallel sysplex, across multiple operating system images (Adabas Cluster Services).
- Can *monitor several databases (DBIDs)* and provide reports for each.
- Can be configured in a straight-forward, uniform, and consistent way under all targeted operating system platforms in a *client/server environment*.
- *Collects, reports, and controls data centrally*. The hub is a true data collection point unencumbered by dependencies on internal operations and structures of the Adabas nucleus.
- Provides unencumbered Adabas operation. The hub *reduces overhead* by eliminating the Review subtask from the Adabas address space, partition, or region.
- Allows you to *customize command logging*. Adabas Review can physically log selected commands and/or buffers based on parameters within the Adabas Review report definitions. Each command log corresponds to a particular report definition, and contains only those records that have been selected as meaningful for the end user.
- Allows you to *customize analysis tools*. Adabas Review reports may be customized to perform functions such as cost accounting and application debugging.
- *Collects history data*. Data collected for Adabas Review reports can be written to the Adabas Review repository at specified intervals and stored as history data. The data can be used in trend analysis.
- Provides *more than 40 sample report definitions*, which may be used without modification, or edited to suit specific site requirements.
- The Edit Report function provides *interactive screens* to help you create a report or edit an existing one.
- Supports a *wide range of data types*. More than 340 data fields are available for specifying the types of data to be collected by Adabas Review.

- Provides *user-defined data fields*. Adabas Review provides a mechanism for defining up to five data fields to meet user specifications.
- *Data can be retrieved online*. Adabas Review reports are executed online or in batch mode. Data collected by Adabas Review reports may be viewed online.
- Provides an *online display of Adabas availability*. The Available Availability (AA) function displays a profile of the databases that are monitored by Adabas Review.
- Provides *access to Adabas Online System (AOS)*, an online Adabas maintenance product.

Index

A

- Adabas
 - interregion communication, 5
 - multiple nuclei, 31
 - user buffer extension, 9
- Adabas Cluster Services, 31
 - use with, 6
- Adabas Online System, 14, 32
- Adabas Parallel Services
 - use with, 6
- Adabas Review
 - allocating storage for, 16
 - defined, 4
 - purpose of, 4
 - structure of, 4, 6
 - diagram, 7
 - summary of features, 31
 - supplied reports, 31
- ADACLX
 - command log extension module, 7
- ADALNK, 5
- ADALOG
 - command logging module, 7
 - interface client component, 9
- ADAMPM, 5
- ADAREV
 - hub server component, 6
- ADARUN parameters
 - NAB to size the attached buffer, 16, 17
 - NC to size the command queue, 16
 - REVIEW=, 9
- ADARVU
 - interface client component, 7, 9
 - transferring data to the hub, 7
- ADASVC, 5
- Autostarted reports, 14

B

- BUFFER-SEGMENTS parameter, 18

C

- Client / server architecture, 5
 - diagram, 7
- Client/server approach, 31
- Command log processing, 31

- Commands
 - types used with reports, 23
- Components
 - hub server, 6
 - diagram, 7
 - interface client, 7
 - diagram, 7

D

- Data types supported, 31
- Database fields
 - user-defined, 32
- Databases
 - multiple, 31

G

- GENAUTO command, 14
- GENCARD command, 14

H

- History data, 31
 - collecting, 12
- Hub concept, 31
- Hub mode
 - collecting data in, 10
 - diagram, 10
 - command log processing, 9
 - requirements for use, 5
- Hub server
 - advantages of, 6
 - components, 6
 - diagram, 7
 - defined, 6
 - location, 5
 - storage requirements, 16
 - attached buffer, 17
 - command queue, 16
 - transferring data to, 7

I

- Interface client
 - components, 7
 - diagram, 7
 - defined, 7
 - location, 5

transferring data to the hub, 7

L

Link routine exits, 9

List functions

for reports, 27

Local mode

collecting data in, 4

interactive and batch execution modes, 4

M

MAXSTORE

report parameter, 19

Multiple databases

monitoring, 10

R

RAOAUTO, 14

hub server component, 6

RAOSHIST, 12

hub server component, 6

Reports

about, 21

autostarted, 14

batch mode, 22

commands used with, 23

components, 22

database fields

defined, 23

described, 23

display programs

defined, 22

described, 26

purpose, 13

interactive mode, 22

list functions, 27

MAXSTORE parameter, 19

option parameters

defined, 23

described, 25

overview, 22

processing rules

components, 25

defined, 23

described, 24

multiple, 25

report definitions

components, 23

defined, 13, 22

starting, 14

storage for control blocks, 19

storage for data collection areas, 19

storage requirements, 16, 19

supplied, 22

total storage limit (MAXSTORE), 19

user exits, 26

Repositories, 4, 9

defined, 15

multiple, 15

restricted location in hub mode, 15

REVBHUB

hub server component, 6

Review DB

nucleus and subsystems

hub server component, 6

Review reports

about, 21

REVIEW-BUFFER, 9

storage requirements, 16, 18

REVIEWB

hub server component, 9

RVUALT

alternate history data, 12

RVUAUT1, 14

RVUAUT2, 14

RVUEXI parameters

BS2000-FULLSYNCH, 19

CMS-FULLSYNCH, 19

S

SAVE command, 26

SET command, 15

SETFILE command, 15

START command, 22

Storage

allocating above the 16MB line, 16

allocating from MVS subpool 5, 16

MAXSTORE report parameter, 19

requirements, 16

Supplied reports, 22

T

TRANS-PORT, 5

U

User exits

control transfer of data to the hub, 7

report, 26

User profile system

default user profile, 15

maintaining user profiles, 15

purpose, 15

Users

storage requirements for online use, 16, 20

V

VIEW command, 27

W

Work areas

storage requirements, 16, 20