

Adabas Review

Concepts and Facilities

Version 4.9.1

September 2018

This document applies to Adabas Review Version 4.9.1 and all subsequent releases.

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

Copyright © 2018 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

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

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

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

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Document ID: REV-CONCEPTS-491-20200128

Table of Contents

1 About this Documentation	1
Document Conventions	2
Online Information and Support	2
Data Protection	3
2 Summary of Features	5
3 Adabas Review Processing and Components	7
Collecting Data in Local Mode	8
Collecting Data in Hub Mode	9
Collecting History Data	15
Scheduling Reports - Collecting Future Data	15
Processing Data Online Using SYSREVDDB	15
Processing Data in Batch Mode	17
Output Files	18
User Profile System	26
ADARUN Parameters	27
Storage Requirements	27
4 About Adabas Review Reports	33
Detailed vs. Summary Reports	34
Components of a Report	34
Starting and Stopping or Scheduling Reports	41
Autostarted Reports	43
List Report Functions	44
Record Filtering	44
5 About Adabas Review Client Reporting	47
Installation and Activation of Client Reporting	48
Client Reporting Data Collection Process	49
Data Collected for Client Reporting	51
Reporting on Client Reporting Data	52
Adabas Nucleus (PROGRAM=ADANUC) ADARUN REVIEW Parameter Setting	
Considerations for Data Collection Under Client Reporting	52
6 Display Programs Using Software AG Editor	55
Traditional Display Programs vs. Display Programs Using Software AG	
Editor	62
7 Integration with webMethods Optimize for Infrastructure	65
8 Getting Started	67
Accessing Adabas Review Online	68
Using Function Codes	71
Using Adabas Review Commands	71
Using the Online Help System	72
Accessing Adabas Online System (AOS) from SYSREVDDB	73
Accessing Adabas Review Repositories	73
Accessing Technical System Information	74
Ending a Session	80

9 Cost Accounting Example	83
Viewing Data Online	84
Processing Data in Batch Mode	85
Adabas Review Command Response Time Calculation	86
Index	87

1

About this Documentation

■ Document Conventions	2
■ Online Information and Support	2
■ Data Protection	3

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Monospace font	Identifies service names and locations in the format <i>folder.subfolder.service</i> , APIs, Java classes, methods, properties.
<i>Italic</i>	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
Monospace font	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the symbol.
[]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at <http://documentation.softwareag.com>. The site requires credentials for Software AG's Product Support site Empower. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at <https://empower.softwareag.com/>.

You can find product information on the Software AG Empower Product Support website at <https://empower.softwareag.com>.

To submit feature/enhancement requests, get information about product availability, and download products, go to [Products](#).

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the [Knowledge Center](#).

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at https://empower.softwareag.com/public_directory.asp and give us a call.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at <http://techcommunity.softwareag.com>. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

2 Summary of Features

Adabas Review provides a set of tools that allow you to monitor 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.

The major Adabas Review features are listed below. Adabas Review:

- 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 Adabas Review subtask from the Adabas address space, partition, or region.
- Allows you to *customize command logging*. Adabas Review can physically log selected commands 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.
- *Collects client reporting data*. When activated, client reporting data collection will allow you to better determine where command processing time is spent.

- Provides *a series of sample report definitions*, which may be edited to suit specific site requirements. We recommend that you review the settings of supplied reports to make sure they meet your site's requirements before attempting to run them.
- Provides *interactive screens* via the Edit Report function to help you create a report or edit an existing one.
- Supports a *wide range of data types*. More than 600 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 35 data fields to meet user specifications.
- *Allows data to 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.
- Allows *scheduling of reports* for a particular duration over a user defined timeline.

3

Adabas Review Processing and Components

■ Collecting Data in Local Mode	8
■ Collecting Data in Hub Mode	9
■ Collecting History Data	15
■ Scheduling Reports - Collecting Future Data	15
■ Processing Data Online Using SYSREVDB	15
■ Processing Data in Batch Mode	17
■ Output Files	18
■ User Profile System	26
■ ADARUN Parameters	27
■ Storage Requirements	27

Adabas Review runs in:

- *local mode* in the Adabas address space;
- *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; or
- *client mode* as a server in its own address space with a client interface with the TSO/batch, CICS, or IMS TP monitors. Client mode is supported in z/OS environments only.

Read *Software AG Product Support and Requirements*, in the *Adabas Review Release Notes*, for a matrix of supported Adabas versions and other requirements.

Collecting Data in Local Mode

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

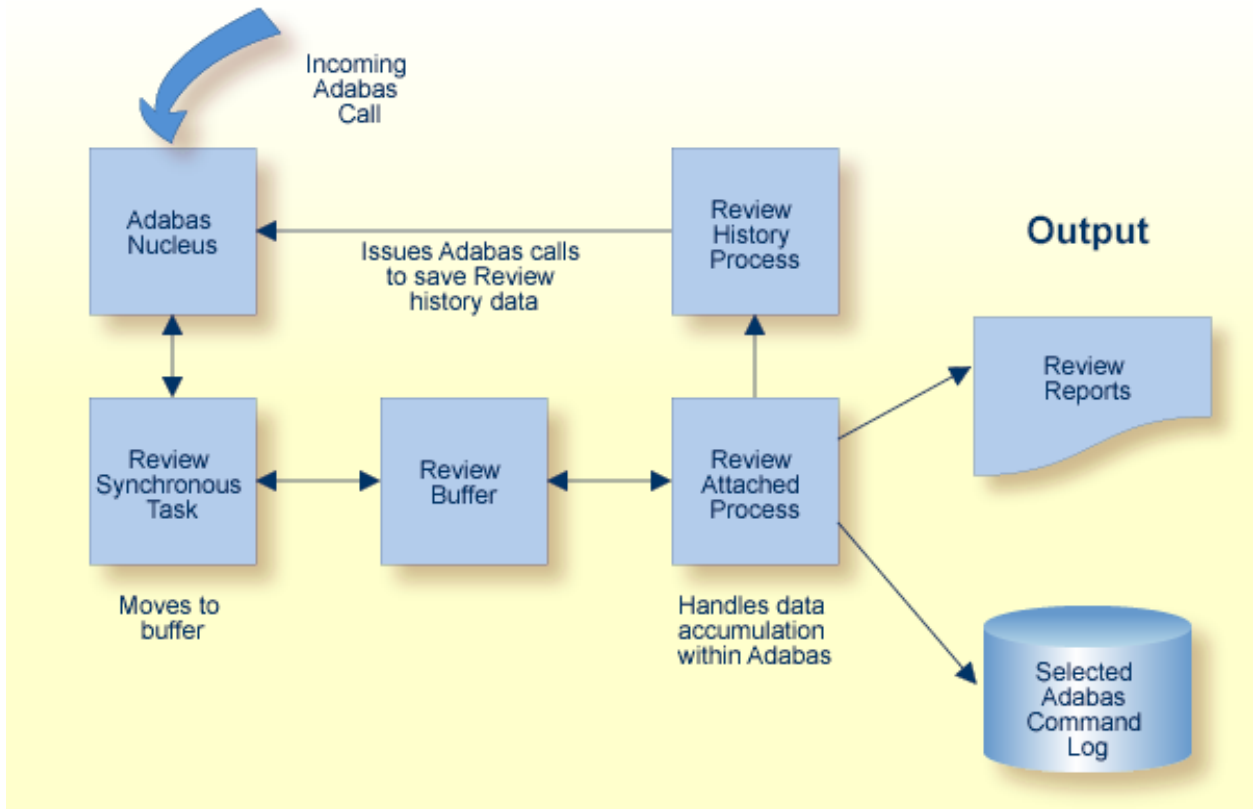
The data collection process is partly accomplished by the Adabas Review processor. The data is collected by providing code (RAOSLOCL) that runs as an extension to ADALOG. Adabas calls this module and passes information about resource usage for each command processed by the Adabas nucleus.

This module, in conjunction with the Adabas Review processor and an intermediate Adabas 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.

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



In local mode, Adabas Review job control statements should be added to the Adabas nucleus startup job.

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: ADALNK, Trans-port, 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.

In hub mode, Adabas Review job control statements should be added to the Adabas Review hub startup job.



Note: Trans-port is available only in z/OS environments.

- [The Server Hub](#)
- [The Client Interface](#)
- [Interface Calls](#)
- [Example Client/Server Environment](#)
- [Command Log Processing](#)
- [Example Hub Mode Data Collection Process](#)

The Server Hub

The Adabas Review hub is a centralized data collector and reporting interface that combines proven components of Adabas and Adabas 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 Adabas 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 Adabas Review data calls and requests;
- REVHUB, a module to establish and maintain the environmental settings for Adabas Review; and
- the Adabas 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 Adabas 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 Adabas 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 Adabas 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 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.

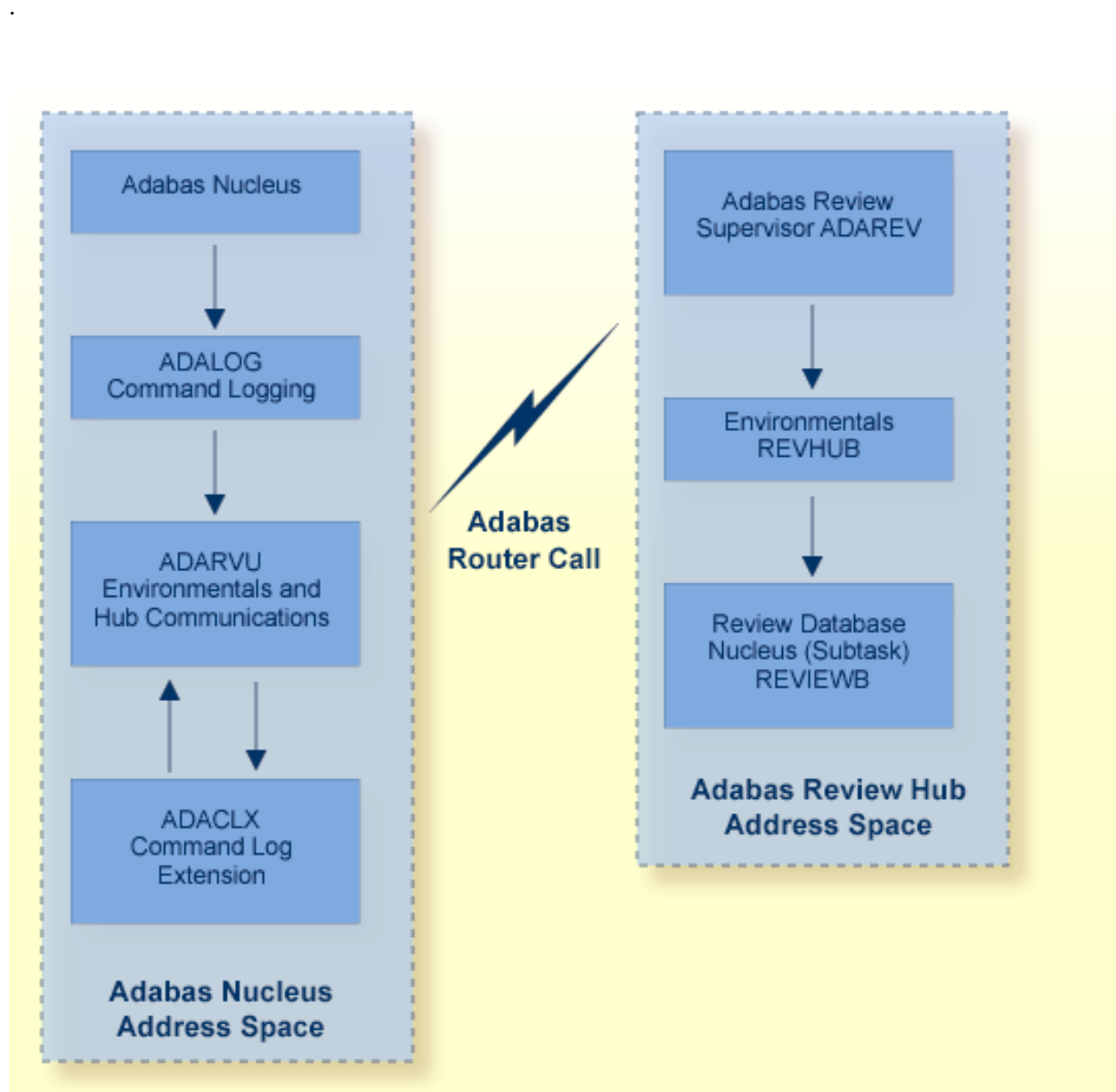
Command log records are normally buffered to reduce the number of calls from the Adabas nucleus to the Adabas Review hub. In low traffic situations, this might lead to a delay of approximately five seconds.

If the communication to the hub results in a non-zero response code (for example response code 151), ADARVU will call user exit 5 if it is available. If a user exit is used, you are given the option to resend the whole buffer (with or without a wait), although parts of the buffer may already be processed. We recommend that you *not* resend the buffer, but generate a message indicating an error has occurred correct the problem. For more information about user exit 5 read *REVUEX5: Adabas Review Hub Event Handler (Adabas Exit 5)* in the *Adabas Review Reference Guide*.

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 the Adabas Review hub (server) through an intermediate REVIEW-BUFFER in the Adabas subtask address space. Note that command log records are examined prior to queuing to determine whether record filtering is active. When filtering is active, only command log records that are filtered are sent to the hub.

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

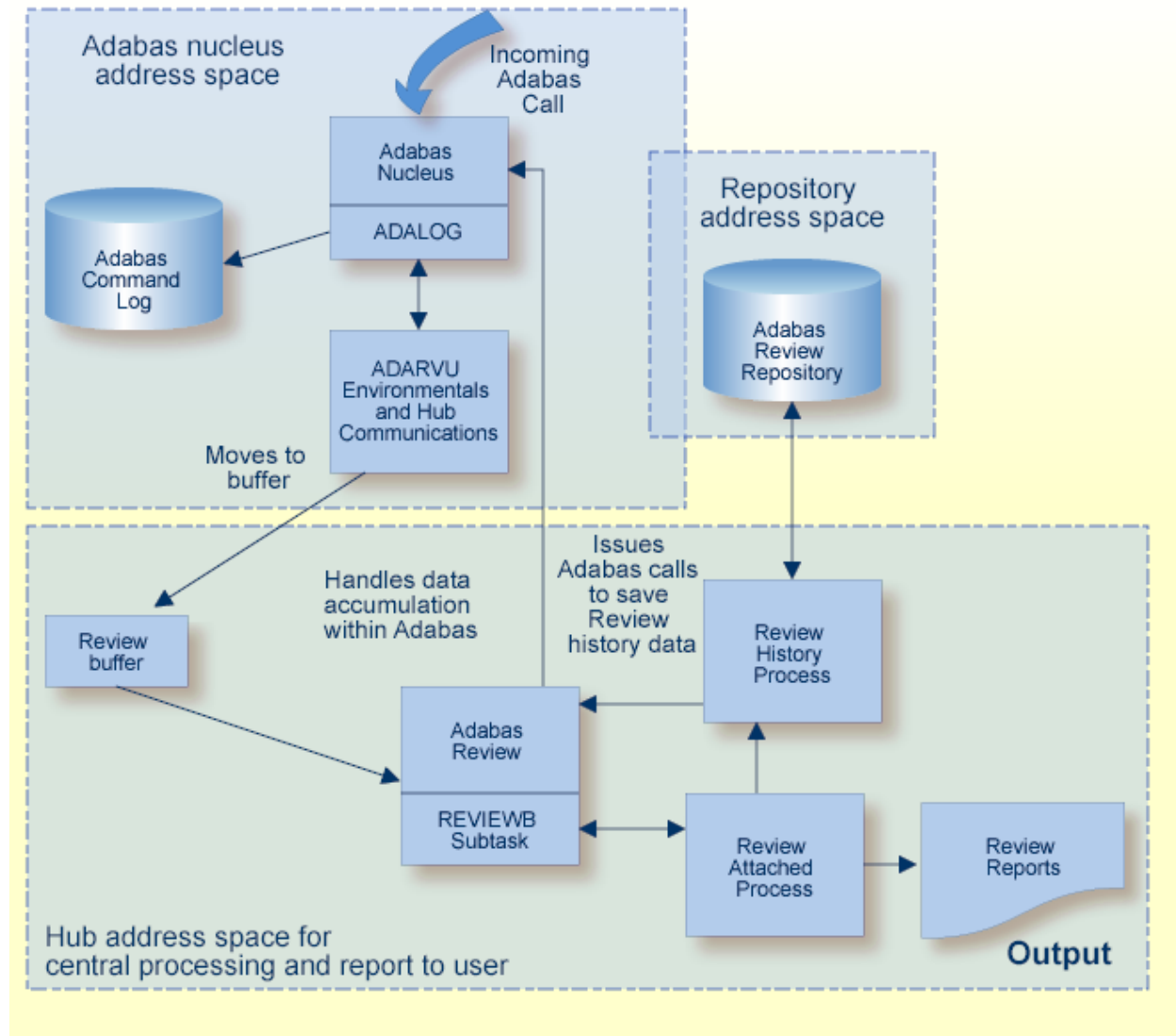
REVIEWB processes the records, accumulating and tabulating various data according to the criteria specified in any user-defined reports that are active.

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.

The format and length of fields stored in the history data is defined in an internal field table in Adabas Review.

z/OS and 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.

Scheduling Reports - Collecting Future Data

Adabas Review uses operating system timer functions to schedule reports. The scheduling parameter may be entered in report definitions as options.

Processing Data Online Using SYSREVD

Adabas Review can be accessed online via a standard Natural application that resides in the Natural library SYSREVD. SYSREVD can be used as an online administration tool. Using commands and screens provided in SYSREVD, you can:

- Perform Adabas Review administration (starting, stopping, and defining reports).
- Access collected monitoring data from summary reports or history data online.
- Display buffer pool information.
- Display active databases.
- Display active maintenance information and environmental information.
- Maintain target objects and user profiles.
- Access Adabas Online System, if it is available.

The format and length of fields shown in online reports is defined in an internal field table in Adabas Review.

For complete information on getting started using SYSREVD, read [Getting Started](#), elsewhere in this guide.



Note: The ADALNK of the TP monitor running SYSREVD must include the REVEXIT exits. If these are not included, some features might not work correctly in SYSREVD.



Important: If an ADALNK batch link routine has been linked or modified by Software AG product modules or user exits, it cannot be used in any application startups of Adabas utility jobs or Adabas, Entire System Server, Adabas Review Hub, or Entire Net-Work nuclei.

The Adabas Review exits include REVEXIT1 (REVEXITB in BS2000 environments) and REVEXIT2. To obtain data for all Adabas Review fields (TP monitor-specific and Natural-specific fields) that are obtained in the Adabas client address space (ADALNK), the REVEXIT1 entry point of the Adabas Review exit REVEXIT and the ADALNK routines from Adabas are used. To obtain data for all client reporting fields, the REVEXIT2 entry point of REVEXIT and the ADALNK routines must also be used.



Note: In BS2000 environments, REVEXIT2 is not available, so the functionality supplied by REVEXIT2 is missing on this platform.

These exits should be linked with ADALNK during installation, with the appropriate Adabas link globals table parameters (LGBLSET). The link globals table parameters specified via LGBLSET identify which exit entry points (REVEXIT1 or REVEXIT2) are called. If the LGBLSET parameter REVIEW=YES is set, then the exit is called by ADALNK before the Adabas call as entry point REVEXIT1; if the LGBLSET parameters RVCLNT and REVHID are also specified, then the exit is also called after the Adabas call as entry point REVEXIT2.

The modules used to perform the linkage between REVEXIT and ADALNK are provided with your Adabas Review installation with names in the format RDBLX_{sys}, where *sys* is a three-character code representing the link environment (for example, MVS for z/OS environments, VSE for z/VSE, COM for Com-plete environments, IMS for IMS environments, and BS2 for BS2000 environments). For more information about performing this linkage, read the appropriate Adabas Review installation documentation.

Processing Data in Batch Mode

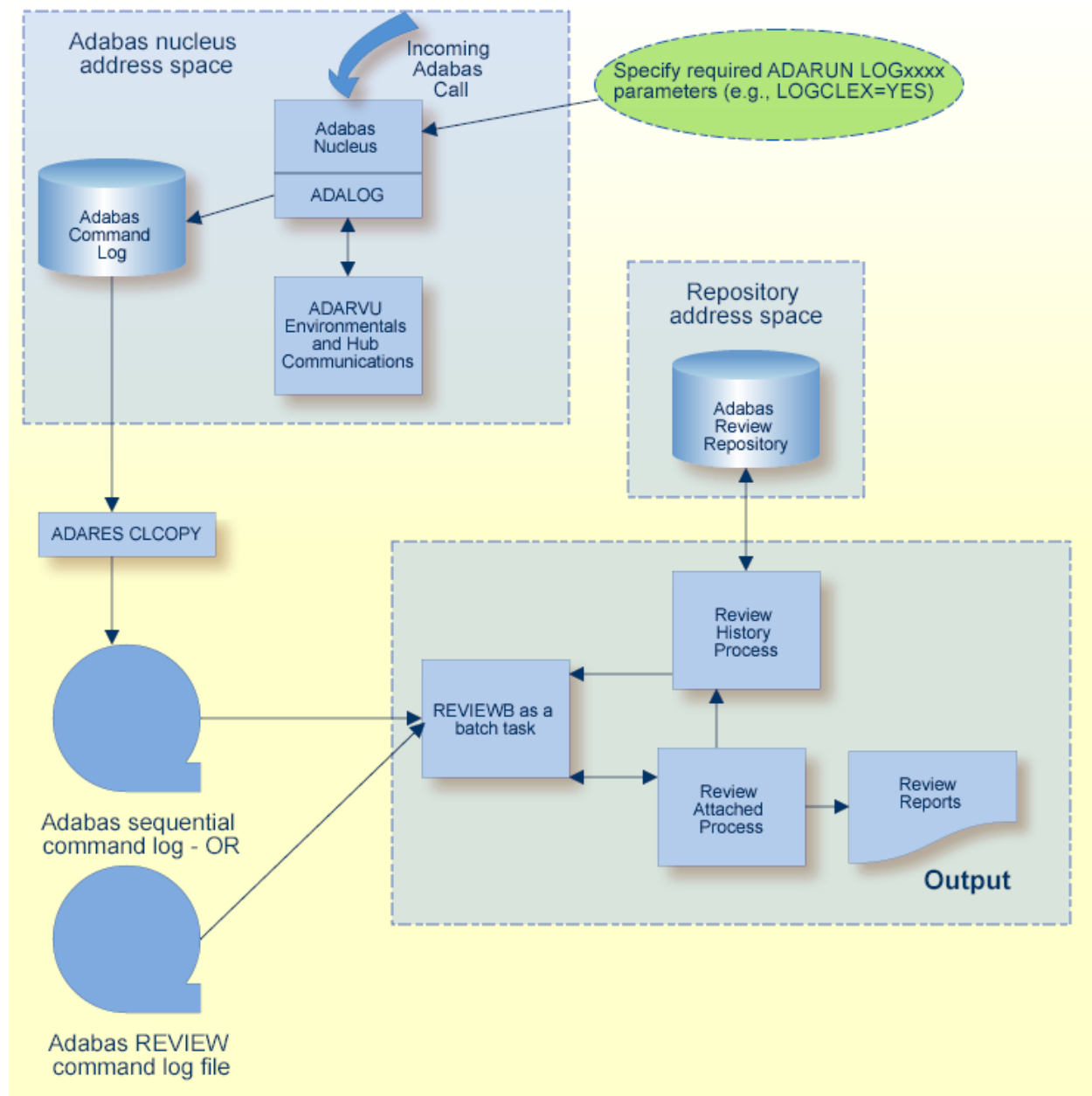
As a batch job, the Adabas Review command log processing routine REVIEWB processes Adabas command log records from a sequential data set. This input can either be created by the Adabas server writing command logging or by the Adabas Review online processor writing a command log file. See [Command log files](#) in the section *Output Files* below.

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. The command log data set must be specified in the Adabas Review startup JCL using the symbolic job name (DD name, DLBL name, or SET-FILE-LINK name) RVUSEQ.

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.

The following diagram depicts batch mode processing. It shows how Adabas creates a command log (a BDAM data set) that must then be copied to a sequential output command log (data set) using the ADARES CLCOPY utility function. This sequential command log is then used as input to Adabas Review in batch mode. The ADARUN parameters of the Adabas nucleus influence the data that is stored in the sequential data set and only data in the sequential data set is shown by Adabas Review via batch processing. For example, if the ADARUN LOGCLEX parameter is set to NO for the Adabas database, no CLEX information is available in the sequential command log and Adabas Review cannot report on the CLEX fields (which include fields in the NAT and TP field categories).



Output Files

A variety of different output files can be produced by Adabas Review. For more information about each of them, select the name in the "Output File" column below:

Output File	Description
Repositories	One or more Adabas files used to store report definitions, historical data, and target definitions.
RVUPRT00 Output Printer	An output printer defined in the Adabas Review startup job that contains all the parameters, input statements, and final statistics for all reports.
RVUPRTnn Output Printers	A series of output printers defined in the Adabas Review startup job that contain detailed or summary report data.
Command log files	One or more log files containing command data and, optionally, buffer data from the detailed reports. Command log files are specific to a given detailed report.
Summary log files	One or more log files containing data from the summary reports. Summary log files are specific to a given summary report.
Raw log files	One or more log files containing raw data from detailed and summary reports. Raw log files are specific to a given detailed or summary report.

Repositories

The Adabas Review repository is an Adabas file used for storing report definitions, historical data, and target definitions. In hub mode, this file must be located on an Adabas database which will be accessed by the hub using normal Adabas router calls. It must also be accessible to the Natural nucleus that is running SYSREVD.

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

The format and length of fields stored in the historical data in the repository is defined in an internal field table in Adabas Review. This information is available to you in the field tables provided in the *Field Reference*, in the *Adabas Review Reference Guide*.

RVUPRT00 Output Printer

This output printer contains the parameters, input statements, and final statistics for all reports. This printer file is updated continuously during the Adabas Review session; final statistics for the session are written when Adabas Review shuts down. The RVUPRT00 statement is the JCL control statement name for the output printer definition in the Adabas Review startup JCL.

RVUPRTnn Output Printers

Both detailed and summary data are written to RVUPRT nn output printers when the PRINT report option is set to "Y" for any report. Each output printer contains the data from the report. RVUPRT nn statements are the JCL control statement names for these output printer definitions in the Adabas Review startup JCL, with values for nn ranging from "01" through "99".

RVUPRT nn output printers are assigned to detailed and summary reports in the following manner:

- A new RVUPRT nn output printer is written for each individual detailed report running with the PRINT report option set to "Y". The numbers of the RVUPRT nn output printers increment as new detailed reports are started.
- A *single* RVUPRT nn output printer is used for *all* summary reports that are running with the PRINT reporting option set to "Y". The number assigned the summary output printer depends on the order in which detailed and summary reports are started.

For example, suppose you have several detailed reports and summary reports defined with the PRINT report option set to "Y". If you start one of the summary reports first, before any detailed reports, the RVUPRT01 output printer will be assigned to collect all summary data from all summary reports. However, if you started two detailed reports before you started one of the summary reports, the RVUPRT03 output printer would be used to collect all summary data from all summary reports; RVUPRT01 and RVUPRT02 would be used to collect detailed data for the two started detailed reports.

RVUPRT Summary Report Output

Data from summary reports is displayed in the field order specified in the report definition. For each field marked with an order number a control break will occur. For calculation fields no control break is performed. See the examples below.


```

REPORT NAME LEVEL BREAK EXAMPLE
REVIEW PERIOD:      2016-07-18  13:47:03 - 2016-07-18  13:50:14
                     SUM          SUM          SUM
      5MI  JOB-NAME CMD      COMMANDS      I/O      THTIME
13:45 BATCH1  L4          1          4      0.000000
13:45 BATCH1  OP          1          0      0.112625
13:45 BATCH1  RC          1          0      0.004375
13:45 BATCH1  **          3          4      0.117000

13:45 BATCH2  L4        2,468        32      0.000000
13:45 BATCH2  OP          2          0      0.401500
13:45 BATCH2  RC          2          0      0.016250
13:45 BATCH2  **        2,472        32      0.417750

13:45 DAEFCO  L3          88        120      9.195125
13:45 DAEFCO  RC          2          0      0.044000
13:45 DAEFCO  S1          4          9      0.822750
13:45 DAEFCO  **         94        129     10.061875

13:45 ***** **        2,569        165     10.596625

13:50 BATCH3  L4        1,234        12      0.000000
13:50 BATCH3  OP          1          0      0.200750
13:50 BATCH3  RC          1          0      0.008000
13:50 BATCH3  **        1,236        12      0.208750

13:50 ***** **        1,236        12      0.208750

***** ***** **        3,805        177     10.805375

REPORT NAME LEVEL BREAK EXAMPLE
REVIEW PERIOD:      2016-07-18  13:47:03 - 2016-07-18  13:50:14
      LOG RECORDS:      3805
      LOG RECORDS SELECTED: 3805

```

```
LEVEL BREAK ON PRINT OUTPUT
REVIEW PERIOD:      2016-07-18  13:50:14 - 2016-07-18  13:50:14

 5MI  JOB-NAME CMD  FNR
13:50 BATCH3   L4   100
13:50 BATCH3   L4  *****

13:50 BATCH3   RC   100
13:50 BATCH3   RC  *****

13:50 BATCH3   **  *****

13:50 ***** **  *****

***** ***** **  *****

LEVEL BREAK ON PRINT OUTPUT
REVIEW PERIOD:      2016-07-18  13:50:14 - 2016-07-18  13:50:14
  LOG RECORDS:      45
  LOG RECORDS SELECTED: 45
```

Command Log Files

Command log files contain command data and, optionally, buffer data from the detailed reports if the reports are run with their LOG report options set to "Y". This data can be processed at some later time in an Adabas Review batch run. The output of the command log file depends on the report processing rules you specify for the detailed report and includes only parts of the overall Adabas commands being processed. If report processing rules are specified for the report, only those commands that meet the criteria of the rules are written to the command log file; if *no* report processing rules are specified, all commands are logged to the command log file.

An individual command log file is written for each detailed report started. In addition to logging the basic command log record (the ACB or ACBX) and the CLEX (corresponding to the ADARUN LOGACB, LOGCLEX, and LOGUX parameters), you can opt to log the buffers from the commands using the Buffers to Log report options.

Command log files are defined by job control statements in the Adabas Review startup job with control statement names that have prefixes specified by the FILE report option. When Adabas Review is distributed and for all new detailed reports you define, this prefix is "RVDET". It is necessary to create unique prefixes for every detailed or summary report. If this is not the case an appropriate error message will be printed when the second report with the same file definition starts and this report will then be started without logging. The job control statements for these command log files must be defined in the Adabas Review startup job before Adabas Review starts.

You can define more than one command log file for a report. You specify the number of log files using the NUM OF LOGS report option. When one log file fills, the next log file is used. When the last log file fills, the first log file will be overwritten. To avoid this, you can specify the name of a command log user exit (using the USER EXIT report option) that you've set up to automatically

copy the data from a COMMAND log file when it fills to another location to ensure your data is not lost.

For complete information about log file reporting options, read *Logging Options*, in the *Adabas Review User Guide*.

The field layout and the formats and lengths of fields in command log files match the Adabas command logs and are mapped by the LOREC and LORECR DSECTs found in your Adabas source library.

Summary Log Files

Summary log files contain data from the summary reports when they are run with their SUMMARY LOG report options set to "Y". An individual summary log file is written for each summary report started.

Summary log files are defined by job control statements in the Adabas Review startup job with control statement names that have prefixes specified by the SUMMARY FILE report option. When Adabas Review is distributed and for all new summary reports you define, this prefix is "RVSUM". We recommend you create unique prefixes for every summary report you create to avoid having two summary reports write to the same summary log file. Nevertheless, it is possible that two or more summary reports write to the same summary log file. The parameter definitions of the report which is started first will be used for all reports which are started afterwards. At each close of a summary report this report will write to the file. It is under the user's control to divide the specific records to their specific reports. This can be done using the *Header portion* of the Summary Record Layout. Refer to *Summary Record Layout*, in the *Adabas Review Reference Guide* for details.

The job control statements for these summary log files must be defined in the Adabas Review startup job before Adabas Review starts.

You can define more than one summary log file for a report. You specify the number of log files using the NUM OF LOGS report option. When one log file fills, the next log file is used. When the last log file fills, the first log file will be overwritten. To avoid this, you can specify the name of a summary log user exit (using the LOG FULL EXIT report option) that you've set up to automatically copy the data from a summary log file when it fills to another location to ensure your data is not lost.

For complete information about log file reporting options, read *Logging Options*, in the *Adabas Review User Guide*.

The format and length of fields stored in the summary log files is described in tables in the *Field Reference*, in the *Adabas Review Reference Guide*.

Raw Log Files

Raw log files are sequential files that contain data from detailed and summary reports when they are run with their WRITE RAW DATA report options set to "Y".

Raw log files are defined by job control statements in the Adabas Review startup JCL with control statement names that have prefixes specified by the RAW DATA FILE report option. When Adabas Review is distributed and for all new detailed or summary reports you define, this prefix is "RAWDT" for detailed reports and "RAWSM" for summary reports. It is necessary to create unique prefixes for every detailed or summary report. If this is not the case an appropriate error message will be printed when the second report with the same file definition starts and this report will then be started without logging. The job control statements for these raw log files must be defined in the Adabas Review startup job before Adabas Review starts.

You can define more than one raw log file for a report. You specify the number of log files using the NUM OF LOGS report option. When one log file fills, the next log file is used. When the last log file fills, the first log file will be overwritten. To avoid this, you can specify the name of a raw log user exit (using the SWITCH EXIT report option) that you've set up to automatically copy the data from a raw log file when it fills to another location to ensure your raw data is not lost.

For complete information about log file reporting options, read *Logging Options*, in the *Adabas Review User Guide*.

The format and length of fields stored in the raw log files is described in tables in the *Field Reference*, in the *Adabas Review Reference Guide*.

This section covers the following topics:

- [Raw Log File Format](#)
- [Raw Log Header Format](#)

Raw Log File Format

The format of the data in the raw file records matches that of the report data.

All raw data records are variable-length records and have a four-byte indicator (the length word) at the beginning of each record in the standard format *nn00*, where *nn* is the length of the record, including the length word, in binary format. Data in the record may be delimited by a character of your choosing if you have the two parts of the report option Data Dlmtr set properly. To delimit the data, set the first part of the report option to "Y" and specify an EBCDIC character to be used as a delimiter for the second part of the report option. For more information, read *Raw Log Options*, in the *Adabas Review User Guide*. If you choose not to use a delimiter, all the fields in the record are strung together in the raw data record with no separator in between them.

Each raw log file begins with a raw header if the report option Write Header is set to "Y". This header shows how subsequent raw data records are defined. If used, it describes the fields in the data record. The format of the raw header is described next in [Raw Log Header Format](#).

Raw Log Header Format

The raw log header record contains the following items, in the following order:

1. A four-byte record length field (RDW) in *nn00* format, where *nn* is the length of the record, including the length word, in binary format.
2. The six-byte string "HEADER".
3. A six-byte field providing the Adabas Review version that produced the raw data.
4. A 32-byte field providing the name of the report used to generate the raw data.
5. A one-byte field indicating whether the report is a detailed (D) or summary (S) report.
6. A two-byte field specifying the delimiter characteristics for the data in the file. The first byte indicates whether the data in the raw data record is field-delimited. A value of C'Y' indicates that there are delimiter characters separating field data in the output record. If this field is set to C'Y', the next byte contains the value of the EBCDIC character used as the delimiter character. This character and these settings are specified in the report options for the report. For more information, read *Logging Options*, in the *Adabas Review User Guide*.
7. Four bytes reserved for future use by Software AG.
8. The actual field descriptors. These repeat for as many fields as are defined in the raw report, up to a maximum of 20 fields. The field descriptor information is formatted as follows:

Byte Offset	Field Length	Description
0	1	The delimiter character for the field descriptor fields in the header record. This is always C', '.
1	8	Report field name with trailing blanks
9	1	C', '
10	1	Field class indicator C'A' -- Account field – break indicator in summary reports C'C' -- COST field C'M' -- MIN (minimum) field C'P' -- PCT (percent) field C'R' -- RATE field C'S' -- SUM (summary) field C'V' -- AVG (average) field C'X' -- MAX (maximum) field
11	1	C', '
12	1	Data type for the field; C'C' for alphanumeric character or C'B' for binary.
13	1	C', '
14	2	Length of the field in binary format with leading zeroes. All numeric output fields are 8-byte binary fields.
16	1	C', '

Byte Offset	Field Length	Description
17	1	Number of decimal places. Note: The value of this field for character fields will be a blank.

User Profile System

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

Adabas Review provides two default Natural profiles that provide access to SYSREVDDB for users who do not have a profile defined for them. Both profiles have administrative privileges when you first access Adabas Review, providing unrestricted access to Adabas Review functions.

- One of the supplied user profile is called "DEFAULT".



Note: The DEFAULT user profile initially provides unrestricted access to Adabas Review functions. Software AG recommends that you first create a new user profile for the system administrator and other privileged users; then modify the DEFAULT profile so that it conforms to the needs of the majority of users. If the default profile is customized so that the access rules meet the needs of the majority of Adabas Review users, the need for individual user profiles can be eliminated.

The DEFAULT profile cannot be deleted. In addition, Adabas Review will not allow you to turn the DEFAULT profile or any administrator profile into a non-administrator profile if another administrator profile is not available.

- The other user profile is saved under the name of the user ID of the user initializing the system (the user who entered MENU).

User profiles can be created for new users, changed for existing users, and purged when no longer required. User profiles can be set up so users have access only to specific DBIDs. 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.

You can also set up group profiles, based on the groups set up by your Natural Security administrator. Natural Security is only required if you want to use group IDs, but not otherwise.

When a user logs on, Adabas Review looks for a group profile first and then for an individual user profile. It uses the first profile found (group profiles take precedence over individual user profiles). If neither a group nor individual user profile is found, the provided default profile is used.

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, read *ADARUN Parameters for Adabas Review*

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 environments, 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](#)
- [Storage for the REVIEW-BUFFER](#)
- [Storage for Reports](#)
- [Storage for Sorting of Summary Reports](#)
- [Storage for Online Users](#)
- [Storage for Work Areas](#)

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

Storage for the Hub

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

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

- [Command Queue](#)
- [Attached Buffer](#)

Command Queue

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

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

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

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

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

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

Example

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

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

The Command Queue and Trans-port (z/OS only)

When using Trans-port in z/OS environments, 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 (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.

The Attached Buffer and Trans-port (z/OS only)

When using Trans-port in z/OS environments, 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 512 bytes. When a value for `BUFFER-SEGMENTS` is specified, the total storage indicated by the specification is split in half between 4K buffers and 32K buffers.

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

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

For 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.

An index is needed for the administration of this pool. The size of this index depends on the size of the REVIEW-BUFFER. For each 128 bytes, one byte is needed for the administration index. If $\text{BUFFERS-4K}=256$, there is an allocation of $256 \times 4K = 1,055,744$ bytes; thus this will need an additional 8,248 bytes ($1,055,744 / 128$) for the administration index.

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.

Each collected record requires storage equivalent to the length of the account record plus two (2) additional bytes. A record cannot be split over the 2K line.

You can see the actual length of the account record used for a specific report by entering the `IN` command on the screen in the online administration tool, `SYSREVDDB`. For more information, read *Displaying Active Report Information*, in the *Adabas Review User Guide*.

Total Storage Limit

The total storage allocation for a report is limited by the `MAXSTORE` report parameter in batch reports and by the "Max K" option in online reports. When the total storage allocation for a report is equal to the `MAXSTORE` or "Max K" value, the report is marked as inactive and stops accumulating data. When a report is purged, all storage associated with the report is deallocated.

For more information about the "Max K" online option, read *General Report Options*, in the *Adabas Review User Guide*. For more information about the `MAXSTORE` batch report parameter, read *REPORT Statement*, in the *Adabas Review User Guide*.

Storage for Sorting of Summary Reports

To sort the result of the data collected by a summary report, a sort area space is allocated. This space is used by all reports. Its size depends on the size of the collected data, which depends on the size of the account records and the number of account records. The maximum size for a specific report is set by the `MAXK` parameter for that report. If a larger area is needed by a new report, the existing area is deallocated and the larger area is allocated.

Once the summary report sort area space is allocated for the largest report, it is not deallocated.

Storage for Online Users

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

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

Storage for Work Areas

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

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

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

4

About Adabas Review Reports

■ Detailed vs. Summary Reports	34
■ Components of a Report	34
■ Starting and Stopping or Scheduling Reports	41
■ Autostarted Reports	43
■ List Report Functions	44
■ Record Filtering	44

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.

It retrieves performance information about Adabas by collecting data in the form of reports, which 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.

Adabas Review provides commands to:

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

Detailed vs. Summary Reports

Adabas Review can produce detailed and summary reports.

- For detailed reports, data is collected continuously while the database is active. Such reports contain a complete recording of the commands processed while the database is active. Detailed reports cannot be viewed online, because the data is not stored in memory. It is directly processed and for example written to an output dataset. However, when fields are defined, a display program is generated that may be used to view history data online if the report collects history data. Results of detailed reports are printed at database termination.
- For summary reports, data in the report is summarized by a particular field (for example, by command, user ID, or program). Results can be viewed online. The report data is stored in memory as long as the report is not stopped. For more information see [Starting and Stopping Reports](#). Depending on the report definition the data of multiple Adabas command log records will be collected and aggregated in one record, such a record is called an account entry.

Components of a 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 *Natural display program* is a Natural program that specifies the appearance of the report output, whether viewed online or downloaded to a personal computer (PC). A detailed report with LOG=Y may be defined without fields. Hence, no display program will be generated.

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.

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 should be modified to meet the requirements of your particular site. Detailed information about supplied reports is provided in *Supplied Report Reference*, in the *Adabas Review Reference Guide*.

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.
- *Report 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, whether a report may be scheduled to start in the future, 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:

20:35:15

A D A B A S - R E V I E W
Edit Report

YYYY-MM-DD
Target=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 Exit SOnly Save Start - + Sched 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:

```

20:24:56                A D A B A S  -  R E V I E W                YYYY-MM-DD
                        Report Processing Rules                    Target=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 SOnly Save Start -   +   Sched      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.



Note: One of two different Report Options screens may appear in Adabas Review, depending on whether you have selected a summary report or a detail report. The following sample is for a summary report.

```

10:50:25          ***** R E V I E W *****          2017-09-28
                   Options for summary report: COMMANDS BY HOUR          Target: 00296

+-- Summary Rpt Options ----- Summary Logging ----- History -----+
| AutoStart ..... Y      Summary Log ..... N      History ..... N |
| Break ..... Y          Summary File ... RVSUM     History Int ... ____1 |
| Wrapping ..... N      Num of Logs ..... _2      History DBID ... 20000 |
| Print ..... Y          Log Size .... __99999     History FNR .... __149 |
| Rstrt/Intrvl Y ____1   Log Full Exit _____  History SVC ..... 249 |
| Max Restarts .. 999999                                     |
| Max K ..... ____8     -- Raw Data Output ---     --- Summary Exit ----- |
| ADALimit ..... ____1  Write Raw Data ..... N     Exit Name ... ____ |
| Display By .. SORTED  Write Header ..... N      Cmd ..... CL |
| Entries ..... 999999  Data Dlmtrs ..... N #      Sum ..... Y |
| Limit ..... 99999999  Raw Data File .. RAWSM |
| Page/Line .. _55 / 133 Num of Files ..... _2 |
|                               Log Size ..... __9999 |
|                               Switch Exit . ____ |
+-----+-----+-----+
Command: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit      Accpt      +      Dflts      Menu

```

The second Report Options screen (press PF8) applies to summary and detail reports.

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

Report User Exits

- 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.

For more information about these exits, read *REVUXDET: Report Exit for Detailed Reports* and *REVUXSUM: Report Exit for Summary Reports*, in the *Adabas Review Reference Guide*.

Natural Display Program

When a report definition is saved, Adabas Review automatically generates a unique Natural display program, called the Adabas Review report display program. 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.

There is a second way of generating a Natural display program: depending on the setting of the `DISPLAY` parameter in the `CONFIGDB` file, a traditional display program will be generated (`DISPLAY=BASIC`) or a new display program will be generated (`DISPLAY=EDITOR`). The new display program uses the Software AG Editor for browsing data and is generated in structured mode. For new display programs it is also possible to re-format (extend or reduce) the width of output columns. Both display programs may exist in parallel. You can run this display program:

- 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 an Adabas 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).

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

The traditional display program is executed when the `VIEW` command is issued to display the report results online. The new display program using the Software AG Editor is executed when the `VIEWX` (`VX`) command is issued.



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.



Note: No display program will be generated when a detailed report with `LOG=Y` will be defined without fields.

Starting and Stopping or Scheduling Reports

Reports can be run in either interactive or batch mode:

- In *interactive mode*, Adabas or the client address space passes information to Adabas Review about resource usage for each command processed.
- In *batch mode*, Adabas Review runs as a batch job that processes sequential Adabas or Adabas Review command log data sets.

Once the report definitions are edited and saved, the report definition is written to the Adabas Review repository file and the reports can be started. Starting a report tells the Adabas Review data collection process to start collecting data based on the report definition parameters. When scheduling data is defined and marked as active, a report may be scheduled, too.

In interactive mode, the reports can be started or scheduled with:

- the START or SCHEDULE command executed in SYSREVDDB or
- via Batch Natural or
- via Review Online functions from within Natural Programs or
- automatically at startup of Adabas Review as *Autostarted Reports*.

In batch mode, reports are started or scheduled via parameter statement submitted in the job stream.

Detailed reports do not accumulate data. The data is directly saved to a file (RVUPRT, Raw or command log files or the history file); therefore, detailed reports cannot be viewed online.

Summary reports do accumulate data. If the report is stopped, the data is processed differently by Adabas Review, depending on what caused the report to be stopped and on the setting of the report options. If the data is kept, it may be stored in RVUPRT, the Raw or summary log file, or the history file.

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

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

Reports can be stopped using the SYSREVDDB CL (close), PS (purge), RF (refresh), or SU(spend) report commands. Reports stop automatically if one of the following report option limits is set and exceeded:

- Max K
- Entries
- Limit
- Intrvl or History Int

Data is accumulated and written when:

- The report is stopped using the CL command;
- The report stops automatically because of an interval event (Intrvl report option);
- The report has the RESTART=Y option specified and stops automatically because the maximum storage limit (MAX K report option) was exceeded; or
- Adabas Review terminates. Adabas Review terminates in local mode if the nucleus is stopped via the ADAEND or HALT operator commands or the hub is stopped via the ADAEND operator command.

Such a portion of a data collection is one interval for a report.

Scheduled reports are queued in the so called wait queue and become active or waiting according to the scheduling parameter. Once the last defined interval has been executed, a scheduled report remains in the wait queue until it will be deleted using the PS (purge) command.



Note: In local mode, data is written with the next Adabas command log record collected by the Adabas Review system.

Data is *not* written when a report is stopped using the PS or RF commands.

When one of the following situations occurs, the data is kept in storage and may or may not be written, depending on the next actions you take:

- The report is stopped using the SU command;
- The report stops automatically because the maximum number of entries that the report can maintain or report has been reached (ENTRIES or LIMIT report option); or
- The report has the RESTART=N option specified and stops automatically because the maximum storage limit (MAX K report option) was exceeded (the report status is set to "I" in this situation).

Autostarted Reports

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

For more information about running autostarted reports, read *Running Autostarted Reports* in the *Adabas Review User Guide*.

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

Adabas Review record filtering can decrease the number of command log (CLOG) records passed from the database to Adabas Review. When filtering is active, only those records needed for report processing will be passed to Adabas 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 Adabas 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 Adabas 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 Adabas 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 Adabas Review reports are non-filtering reports.
- CLOGLAYOUT=8 is required for record filtering.
- Filtering will not be active when running with LOGCLEX=YES.

Adabas Review Fields Eligible for Record Filtering

The following Adabas Review fields can be used in rule processing in order to activate Adabas Review Record Filtering: ACBUSER, ADADURA, ASSOIO, CALLPGM, CID, CMD, CMDTYPE, DATAIO, DBID, FILE, ISN, ISNQ, JOBNAME, NATAPPL, NATPROG, NATSTMT, NATUID, NUCID, RSP, STEPNAME, TOTDURA, TPUSERID, TRANSID, and WORKIO.

The use of any other field in rules for active reports will cause filtering to be stopped. For detailed information about particular fields, see the *Field Reference*, provided with this documentation.

5

About Adabas Review Client Reporting

■ Installation and Activation of Client Reporting	48
■ Client Reporting Data Collection Process	49
■ Data Collected for Client Reporting	51
■ Reporting on Client Reporting Data	52
■ Adabas Nucleus (PROGRAM=ADANUC) ADARUN REVIEW Parameter Setting Considerations for Data Col- lection Under Client Reporting	52

Client reporting allows you to collect client data you can use to determine where command processing time is spent. Client reporting is currently provided for Adabas Review hubs only in z/OS environments using the TSO, CICS, and IMS monitors; it is not currently supported in Com-plete. Client reporting is intended as a diagnostic tool and can be activated and deactivated as needed.

Data for client reports is normally collected in the client address space from the client as well as the Adabas data server. Note that data for client reports can be collected when Adabas Review is running either local or hub mode, but can only be transferred to and reported from a hub.



Caution: The process required to collect client data increases the overhead of your system. We therefore recommend that you activate client reporting only for as long as it takes to diagnose a problem. We do not recommend that client reporting be activated all the time.

This chapter covers the following topics:

Installation and Activation of Client Reporting

Client reporting is currently supported for Adabas Review hubs in z/OS environments and only if ADARUN PROGRAM=USER. For complete information on how to install client reporting, read *Installation Phase 2*, in the *Adabas Review z/OS Installation Guide*. Installation of client reporting support involves performing the following tasks at some point in the Adabas Review installation:

- You will need to set two new LGBLSET macro keywords, RVCLNT (used to indicate whether client reporting is allowed for the hub) and REVHID (which specifies the hub ID). New globals tables must be assembled with these keywords.
- The new globals tables must be linked with the appropriate Adabas Review exit modules (which include REVEXIT2).
 1. The Adabas Review exit modules are RDBLXMVS (batch/TSO), RDBLXCIC (CICS), and RDBLXCOM (Com-plete). These modules contain the batch/TSO, CICS, and Com-plete components of Adabas Review that will be used with the Adabas Version 8 CICS link routine.
 2. While client reporting is active, the REVEXIT2 link routine will collect data and send it to the preferred Adabas Review hub after each Adabas call.

Once client reporting is activated, client data collection occurs for each Adabas call from within the ADALNK routine where it has been activated.

- Refer to the *Adabas Review Release Notes* for information about compatibility with Adabas versions.
- If Adabas Review is installed with its CICS link routine, SYSREVDB (which is used to activate and deactivate client reporting) must be running in the same CICS and using the same task-related user exit (TRUE) that will process client data.

Status messages are provided during installation indicating whether or not client reporting is supported by the system. After installation, if client reporting is supported by the CICS system,

client reporting state messages are produced indicating when client reporting is activated or deactivated.

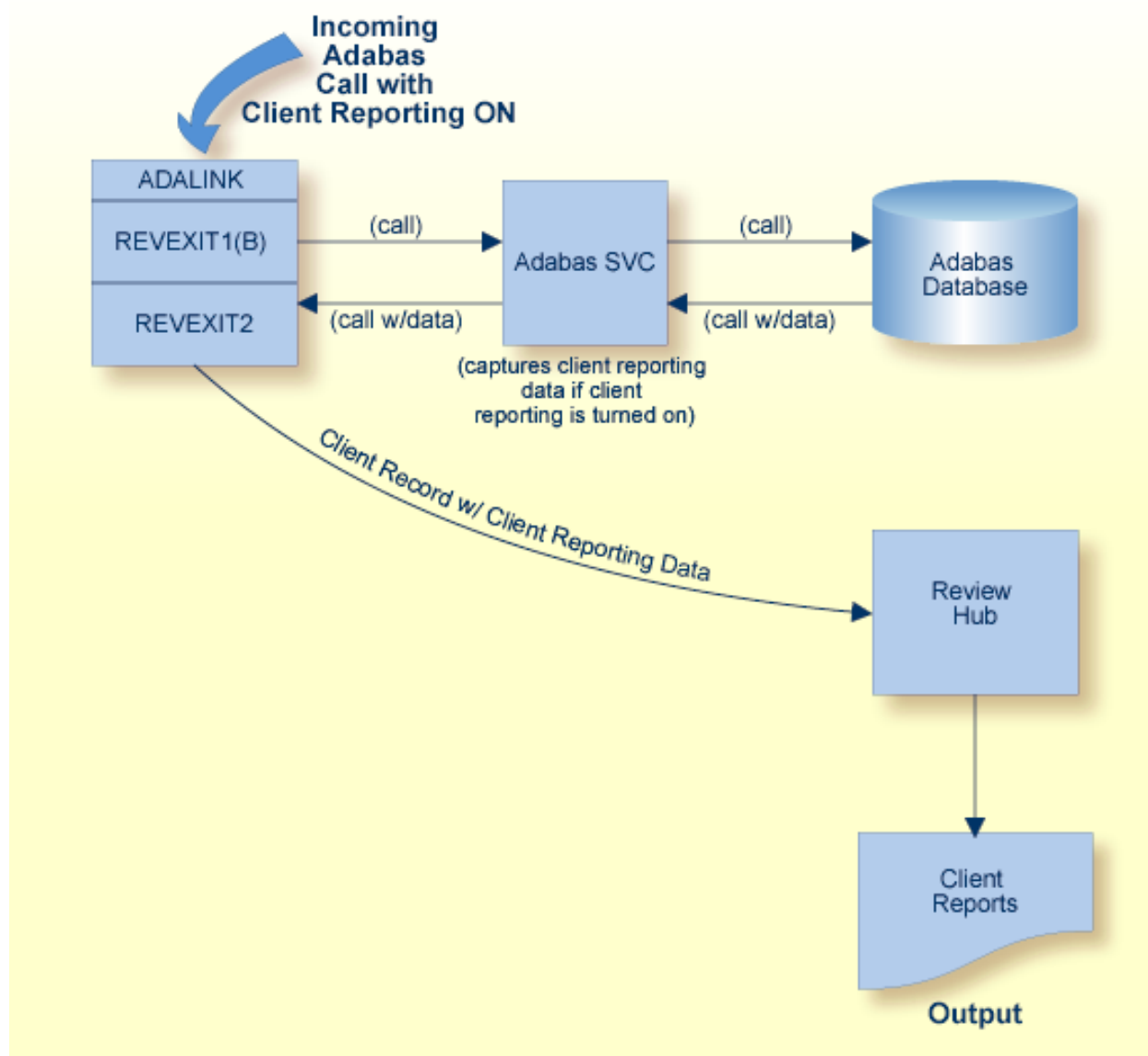
Once client reporting is allowed, it is not automatically activated. Instead, you must manually activate it in one of the following ways.

- Use the Client Management screen in SYSREVDDB to activate it online.
- Set the ADARUN RVCLIENT parameter to "ACTIVE" to activate it when you want to run client reports in batch environments (ADARUN PROGRAM=USER).
- Call the batch module REVCLRP, specifying its RVCLFUNC parameter as "ON" to activate it within your own application program.
- Call the Natural Batch program SETCM with MODE=ON to activate it.

Activate and deactivate client reporting data collection as needed by your installation. To verify if client reporting is turned on correctly please review any LNKRVX* or RVCL* messages appearing on the console log. For more information, read *Managing Client Reporting*, in the *Adabas Review Administration Guide*. Because of the system overhead that can occur while client reporting statistics are collected, we do not recommend that you leave it activated for very long.

Client Reporting Data Collection Process

The following diagram depicts the client reporting data collection process when client reporting has been activated.



When the Adabas call is processed after client reporting has been activated, the Adabas SVC captures the timing data necessary to produce the data stored in the client reporting fields. Then REVEXIT2 collects and sends that data, via Transport, to the preferred Adabas Review hub for the call. The client reporting data is included within a subset of a command log record.



Caution: This process has some impact on the performance of CICS. We therefore recommend that you run client reporting for controlled periods of time as a diagnostic tool only.

Data collection occurs for each Adabas where each Adabas link routine has been prepared to support it.

Data Collected for Client Reporting

Data for client reports is normally collected in the client address space from the client as well as the Adabas data server. Note that data for client reports can be collected when Adabas Review is running either local or hub mode, but can only be transferred to and reported from a hub.

When client reporting is activated, three data fields, specific to client reporting, are collected and included with a subset of the command log record:

- The total client duration time is collected and stored in the CDURA field.. This is the total time (in seconds) in which the client waits for the command to be processed by the server and the time it takes the ADALNK portion of the client to retrieve the command results. CDURA is the sum of the CRCVDURA and CWRKDURA fields.
- The client receive time is stored in the CRCVDURA field. This is the time (in seconds) it takes the Adabas link routine to retrieve a processed command from the server.
- The client wait time, or the time in which the server works for the client, is stored in the CWRKDURA field. This is the time (in seconds) in which the client waits for the command to be processed by the server.



Note: For more information on how duration fields are calculated in Adabas Review, read *Adabas Review Duration Field Derivations*, in the *Adabas Review Reference Guide*.

These specific client reporting fields become available for reports only when client reporting is activated. For complete information about the specific client reporting fields, read about the *Client Reporting Field (CMON)* field category, in the *Adabas Review Reference Guide*.

In addition, a subset of the other field data collected by Adabas Review from the Adabas server is also available for client reports. This includes the data for the Adabas Review fields ADADURA, CQDURA, DBID, JOBNAME, NUCID, ROUTTIME, SEQUENCE, and TOTDURA. If you do not want to collect this data, the appropriate ADARUN parameter of the Adabas server has to be set as described in [Adabas Nucleus \(PROGRAM=ADANUC\) ADARUN REVIEW Parameter Setting Considerations for Data Collection Under Client Reporting](#), elsewhere in this section.

For a complete list of fields available for client reports, read *Fields Available for Client Reporting*, in the *Adabas Review Reference Guide*.

Reporting on Client Reporting Data

Use the Edit Client Report screen in SYSREVDDB to create a client report definition. For more information, read *Maintaining Standard Database and Client Reports*, in the *Adabas Review User Guide*.

In addition, you can use the CLASS parameter of the batch REPORT statement to specify whether the batch report to be generated is a client reporting report. Valid values are "STANDARD" (the report is a standard report) and "CLIENT" (the report is a client reporting report). The default is "STANDARD". The CLASS parameter can be specified on both detailed and summary reports. For more information about the CLASS parameter, read *REPORT Statement*, in the *Adabas Review User Guide*.

Adabas Nucleus (PROGRAM=ADANUC) ADARUN REVIEW Parameter Setting Considerations for Data Collection Under Client Reporting

Data for client reports is normally collected in the client address space from the client as well as the Adabas data server. Note that data for client reports can be collected when Adabas Review is running either local or hub mode, but can only be transferred to and reported from a hub.

Once client reporting is activated (turned on), data is collected for the three specific client reporting data fields: CDURA, CRCVDURA, and CWRKDURA. Data for these fields is collected regardless of the ADARUN REVIEW parameter setting of the databases which are called from within this client program. However, some reporting fields (see the list below) in client reports are affected by the setting of the ADARUN REVIEW parameter:

- To *start*, *run*, or *stop* client reports, the ADARUN REVIEW parameter must not be specified in the Adabas nucleus job. You prepare the Adabas Review client (RVCLNT=YES) specify the hub ID (REvhID) in the ADALNK parameter settings that support client reports.



Important: If an ADALNK batch link routine has been modified to accommodate the needs of an Adabas product extension (such as Adabas Review), it should not be used for the Adabas nucleus or Adabas utility jobs.

- The Adabas nucleus must run with the ADARUN parameter CLOGLAYOUT set to "8" (CLOGLAYOUT=8).
- To *collect* data from the Adabas server in the following Adabas Review fields (while client reporting is turned on), the ADARUN REVIEW parameter specified in the Adabas nucleus job must be set to "LOCAL" or to a hub ID. If it is set to "NO", data is not collected for the following fields:

ADADURA

CQDURA
JOBNAME
NUCID
ROUTTIME
SEQUENCE
TOTDURA

If you do not want to collect the Adabas server data in these fields for a given database, you can either:

- Set the ADARUN REVIEW parameter to "LOCAL" for the given database, and do not define any reports (manually started or autostarted) for this database; or
- Set the ADARUN REVIEW parameter to an invalid hub ID to avoid any standard reporting for this database.

Data will be collected for these additional fields and the overhead in the Adabas server will be minimal.

For more information about the ADARUN REVIEW parameter, read *REVIEW Parameter: Adabas Review Control*, in the *Adabas Review Reference Guide*.

6

Display Programs Using Software AG Editor

- Traditional Display Programs vs. Display Programs Using Software AG Editor 62

In parallel to the traditional display programs, another mode to generate display programs is available. In this mode, the display programs use the Software AG Editor. Both, the traditional display programs as well as the display programs using the Software AG Editor, use the same live or history data.

The Software AG Editor is part of the Natural installation. To be able to use it, an editor buffer pool must be allocated either as a Natural work file or by setting the `EDPSIZE`. For more information see chapter *Software AG Editor* in the *Editors* section of the *Natural for Mainframes* documentation, especially the topic *Operating the Software AG Editor*.

The mode uses the Software AG Editor, because the editor offers an extensive number of browsing functions. The editor supplies standard browsing functions like down, bottom, up, top, left or right. It is possible to fix a number of columns starting with column 1 in the display when scrolling right. Sorting of rows is possible according to specified column and row ranges. All data in the editor may be displayed in hexadecimal format too. For operating on `SYSREVDDB` data, a subset of editor functions is enabled. See the online help when displaying `SYSREVDDB` data for a list of these functions.

```

15:41:59          ***** Review Adabas Tools *****          2016-07-28
                      Help System -Editor Data Display

The Editor Data Display allows you to browse live or history reports.

You can save live reports on the Natural system file for later use.


                                Code Topic
                                ----
                                1  Editor Data Display
                                2  SAG Editor commands
                                .  Exit Help
                                ----

Code .. _

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

```

To differentiate between the modes, there is the new `DISPLAY` parameter in the `CONFIGDB` file. The `DISPLAY` parameter may also be set online using the `DISPLAY` command. With `DISPLAY=BASIC`, display programs will be generated in the traditional way, in reporting mode. The names start with `RD*` (user reports), `SR*` (delivered system reports) or `CR` (client reports). With `DISPLAY=EDITOR`, display programs will be generated using the Software AG Editor and in structured mode. The names start with `RX*`, `SX*` or `CX`.

Like the `REGEN (RG)` command, other commands that affect the generated programs depend on the setting of the `DISPLAY` parameter. These are the `PRINT (HC)`, `DOWNLOAD (DL)`, `EP (Edit Generated Program)` and the `PR (Purge Report Definition)` command. For viewing report data the commands `VIEW (VW)` for traditional display programs and `VIEWX (VX)` for display programs using the Software AG Editor exist in parallel.

This is a sample output of the `APPLICATION FILE FIELD USAGE` system report, using the Software AG Editor, generated by program `SX-00030`:

```

15:39:36                APPLICATION FILE FIELD USAGE                2016-07-28
                        2016-07-28 15:39:32 - 2016-07-28 15:39:35    Columns 010 072
=====>                                Scroll ==>    PAGE
***** ***** top of data *****
00001 -----
00002                      Total          Total          Total
00003 NAT-Appl  File Fld-Name  Num-of-IOs  Commands  CMD-Resp
00004 -----
00005 SYSREVD  0                0            10          0.128500
00006          37 01            0            2          0.010750
00007 ***** *****            0            12          0.139250
00008 UKRPE    0                0            7          0.001750
00009          7 LJ            0            6          0.009750
00010 ***** *****            0            13          0.011500
00011 ***** *****            0            25          0.150750
***** ***** bottom of data *****

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  RVSrt Exit  Updat Rfind      -      +      <=== ===>  Canc

```

```

15:40:08          APPLICATION FILE FIELD USAGE          2016-07-28
                2016-07-28 15:39:32 - 2016-07-28 15:39:35    Columns 025 087
=====>                                           Scroll ==> PAGE
***** ***** top of data *****
00001 -----
00002              Total          Total          Total          Total
00003 NAT-App1    Num-of-IOs      Commands      CMD-Resp      ADA-Dur
00004 -----
00005 SYSREVDDB          0          10          0.128500      0.000047
00006              0          2          0.010750      0.000619
00007 *****          0          12          0.139250      0.000666
00008 UKRPE              0          7          0.001750      0.000133
00009              0          6          0.009750      0.000790
00010 *****          0          13          0.011500      0.000923
00011 *****          0          25          0.150750      0.001589
***** ***** bottom of data *****

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  RVSrt Exit  Updat Rfind      -      +      <=== ===>  Canc

```



Note: In a new display program one line of max. 4000 byte will be redefined according to the fields in the report definition. The generated order depends on the report definition. For program SX-00030 (APPLICATION FILE FIELD USAGE) we have the base fields NATAPPL (#NAT-LOGON), FILE (#ADABAS-FILE), FBFIELDS (#FB-FIELDS) and the sum of IOS (#NUM-IOS-SUM), COMMANDS (#COMMANDS-PER-ACCT-SUM), CMDRESP (#CMD-RESP-SUM) and ADADURA (#ADA-DURATION-SUM).

Since the line defined as #FORMAT-OUT will be put into the editor buffer, you may change the re-definition (order) of the fields. The headers change accordingly without further changes to the program. However, the first field (#OUT-1, here #NAT-LOGON) must stay at the first place, because output groups will be built according to the value of this field. See a sample below:

```

SX-00030.NSP
158 * Besides #OUT-1, the order of the #OUT fields may be changed.
159 1 #FORMAT-OUT (A4000)
160 1 REDEFINE #FORMAT-OUT
161 2 #OUT-1 (A8) /* #NAT-LOGON
162 2 #F1 (A1)
163 2 #OUT-2 (A5) /* #ADABAS-FILE
164 2 #F2 (A1)
165 2 #OUT-3 (A8) /* #FB-FIELDS
166 2 #F3 (A1)
167 2 #OUT-4 (A12) /* #NUM-IOS-SUM
168 2 #F4 (A1)
169 2 #OUT-5 (A13) /* #COMMANDS-PER-ACCT-SUM
170 2 #F5 (A1)
171 2 #OUT-6 (A19) /* #CMD-RESP-SUM
172 2 #F6 (A1)
173 2 #OUT-7 (A15) /* #ADA-DURATION-SUM
174 2 #LINE-ATTR (A1)

SX-00030.NSP
158 * Besides #OUT-1, the order of the #OUT fields may be changed.
159 1 #FORMAT-OUT (A4000)
160 1 REDEFINE #FORMAT-OUT
161 2 #OUT-1 (A8) /* #NAT-LOGON
162 2 #F1 (A1)
163 2 #OUT-7 (A15) /* #ADA-DURATION-SUM
164 2 #F2 (A1)
165 2 #OUT-6 (A19) /* #CMD-RESP-SUM
166 2 #F3 (A1)
167 2 #OUT-5 (A13) /* #COMMANDS-PER-ACCT-SUM
168 2 #F4 (A1)
169 2 #OUT-4 (A12) /* #NUM-IOS-SUM
170 2 #F5 (A1)
171 2 #OUT-3 (A8) /* #FB-FIELDS
172 2 #F6 (A1)
173 2 #OUT-2 (A5) /* #ADABAS-FILE
174 2 #LINE-ATTR (A1)

```

After changing the order of the fields in SX-00030 like in the above sample, the output of the APPLICATION FILE FIELD USAGE report is as follows:

```

11:32:25                APPLICATION FILE FIELD USAGE                2016-07-29
                        2016-07-29 11:32:19 - 2016-07-29 11:32:24    Columns 010 072
=====>                                                    Scroll ==>  PAGE
***** ***** top of data *****
00001 -----
00002                Total                Total                Total                Total
00003 NAT-App1          ADA-Dur          CMD-Resp          Commands          Num-of-IOs
00004 -----
00005 SYSREVDDB          0.000144          0.128500          10                0
00006                0.000973          0.010750          2                0
00007 *****          0.001117          0.139250          12                0
00008 *****          0.001117          0.139250          12                0
***** ***** bottom of data *****

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  RVSrt Exit  Updat Rfind      -      +      <=== ==>  Canc

```

```

11:34:14                APPLICATION FILE FIELD USAGE                2016-07-29
                        2016-07-29 11:32:19 - 2016-07-29 11:32:24    Columns 025 087
=====>                                                    Scroll ==>  PAGE
***** ***** top of data *****
00001 -----
00002                Total                Total                Total
00003 NAT-App1          CMD-Resp          Commands          Num-of-IOs Fld-Name  File
00004 -----
00005 SYSREVDDB          0.128500          10                0                0
00006                0.010750          2                0 01            37
00007 *****          0.139250          12                0 ***** *****
00008 *****          0.139250          12                0 ***** *****
***** ***** bottom of data *****

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  RVSrt Exit  Updat Rfind      -      +      <=== ==>  Canc

```

Furthermore, it is possible for the user to define the output format of fields in a report, for example to reduce the output length for a clearly arranged output according to the user's needs. Also fields may be enlarged to avoid overflow.

For this purpose the Natural text member REDEFS in library SYSREVDDB, if existing, will be read to obtain user defined formats.

REDEFS may be generated with the latest field definitions using the stand-alone Natural program GENREDEF in library SYSREVDDB. To execute GENREDEF, stop SYSREVDDB if running, so that you are at the Natural NEXT prompt in library SYSREVDDB. Enter GENREDEF in the NEXT or Command line and press Enter. It will generate the Natural Text member REDEF_{vrs} in library SYSREVDDB. To activate the just generated REDEF_{vrs} file, rename or copy it to REDEFS. If a Natural Text member with the name REDEFS already exists and you want to keep it, make a backup copy by renaming it or copy it to another library.

Redefinitions may be specified in the section of this file, starting with <FIELDS-REDEF-START> and ending with <FIELD-REDEF-END>. This section will be read when a display program will be (re-)generated. Valid field names with valid format definitions and headers within this section will overwrite the default values. See the documentation at the top of text member REDEF_{vrs} for the format of such an entry. Also, the REDEF_{vrs} file contains the latest SYSREVDDB field definitions as comments.

The format of a variable (A, B, N) may not be changed. Keep in mind, that the minimum column width is the size of the header. Some output columns may be reduced in width just by a shortened header.

User defined fields may not be redefined here.

The fields of APPLICATION FILE FIELD USAGE may be redefined like this:

```
<FIELDS-REDEF-START>
NATAPPL  A8      NAT-App1      /* A8      NAT-App1
FILE      N5      File         /* N5      File
FBFIELDS  A2      FB           /* A2      Fld-Name
IOS        N8      NumOfIOs     /* N12     Num-of-IOs
COMMANDS  N8      Commands     /* N13     Commands
CMDRESP   N4.4    CMD-Resp     /* N12.6   CMD-Resp
ADADURA  N2.4    ADA-Dur      /* N8.6    ADA-Dur
<FIELDS-REDEF-END>
```

The above output example would then easily fit on one page.


```

12:54:22                APPLICATION FILE FIELD USAGE                2017-09-28
                        2017-09-28 12:53:31 - 2017-09-28 12:54:22    Columns 010 054
=====>                Scroll ==>  PAGE
***** ***** top of data *****
00001  -----
00002                Total      Total      Total      Total
00003 NAT-Appl   File FB NumOfIOs  Commands  CMD-Resp  ADA-Dur
00004  -----
00005 SYSREVDDB      0              0          29      0.4640  0.0000
00006 *****      ***** **      0          29      0.4640  0.0000
00007 *****      ***** **      0          29      0.4640  0.0000
***** ***** bottom of data *****

Enter-PF1---PF2---
      Help  RVSrt Exit  Updat Rfind      -      +      Canc

```

For customizing the output, there are further parameters in the CONFIGDB file.

- **HDR-LINE-COUNT:** specifies after how many data lines header lines will be inserted (default: one header per page).
- **FIXED-CHARS:** specifies the number of fixed bytes in the editor area when scrolling left/right (default: fix length of first column).
- **MAX-NO-OF-LINES:** specifies the maximum number of lines loaded into the editor buffer pool (default: 3000).

See also *Modifying Configuration Parameters* in the *Installations and Operations for z/OS*, *Installations and Operations for z/VSE* or *Installations and Operations for BS2000* documentation.

In contrast to the traditional display programs, those using the Software AG Editor read all selected data (live or history) into an editor session, before the data will be displayed. For live data from the LS screen, the data will be updated with PF4 (Updat). The traditional display programs updated the live data with every ENTER. Since all data is available in an editor session, the Redisplay function becomes obsolete. Furthermore, for the DOWNLOAD and the PRINT function and in batch, the data will be loaded into an editor session first. From there the data will be read for output.

It is also possible to save live data on the Natural system file. Special Review data files will be created. These files can be read, listed and purged from within an editor session. They cannot be accessed with other Natural functions for Natural members. To keep live data, alternatively use the DOWNLOAD function. Creating history data is possible too.

16:13:34		***** Review Adabas Tools *****				2016-07-28
Select Member						
C	Member	Type	User	Date	Time	
-	-----	-----	-----	-----	-----	
—	ABS	REV-RESULT	GGS	2016-06-21	12:38:33	
—	RATED1	REV-RESULT	GGS	2016-06-21	11:09:53	
—	RETED3	REV-RESULT	GGS	2016-06-24	12:17:23	
—	XYZ	REV-RESULT	GGS	2016-03-16	15:54:56	
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---						
Help		Exit				

Since the output of data is handled with a Software AG Editor interface and no longer with one WRITE/DISPLAY statement per line, the line size limit of 250 does not apply to display programs using the Software AG Editor. Nevertheless, for compatibility reasons, this line size limit is still active when defining a report. Both, the traditional display programs as well as the display programs using the Software AG Editor, access the same report definitions, the same live data from the Adabas Review nucleus and the same history data on the Adabas Review system file.



Note: Segments of Adabas buffer fields (e.g. IBSEG01) are not displayed in hexadecimal format. To see this data in hexadecimal format use the HEX ON|OFF editor command.

Traditional Display Programs vs. Display Programs Using Software AG Editor

Feature	Traditional Display Programs	Display Programs Using Software AG Editor
DISPLAY	BASIC	EDITOR
Natural Programming Mode	Reporting Mode	Structured Mode
Names of Display Programs	RD-*, SR-*, CR-*	RX-*, SX-*, CX-*
Command to Display Data	VIEW, VW	VIEWX, VX
Other Commands	REGEN ALL (RG), DOWNLOAD (DL), PRINT (HC), EP, PR <i>with DISPLAY=BASIC</i>	REGEN ALL (RG), DOWNLOAD (DL), PRINT (HC), EP, PR <i>with DISPLAY=EDITOR</i>
Sorting Report Results ("Display By" report option)	PF2 (Sort) or SORT command	PF2 (RVSort)
Sorting Report Rows	-/-	SORT Editor command
Download	Columns without delimiter	Variable delimiter, prompted in a popup window

Feature	Traditional Display Programs	Display Programs Using Software AG Editor
Browsing Data	Top (--), Down (+), Redisplay, Left, Right	Top (--), Bottom (++) , Up (-, -n), Down (+, +n), Left, Right, etc.
Manually change data	No	Yes, for example to prepare data for download or print
Save Live Data on Natural System File	No	Yes
Variable Printer Name	No	Printer name prompted, processed by the Natural statement <code>DEFINE PRINTER (1) OUTPUT <printer-id></code> .
Variable Column Order	According to Report Definition	According to Report Definition. Variable Redefinition in Natural Display Program may be changed
Resizing of output columns	No	Yes
Live Data	Updated with every ENTER	Updated with PF4 (Updat)

7

Integration with webMethods Optimize for Infrastructure

Adabas Review key performance indicator (KPI) statistics are provided to webMethods Optimize for Infrastructure. Using the Optimize component of webMethods Optimize for Infrastructure, you can monitor the status of these statistics as well as the overall status of your system. This support is available in version 8.2 (and later versions) of webMethods Optimize for Infrastructure.

The prerequisite Natural version is NAT8.2 SP6 with fix NAT826I004 or an above Natural version.

The key performance indicator statistics that are passed from Adabas Review to Optimize provide information about the Adabas Review system itself and not about the monitoring data collected by Adabas Review. The KPIs that are passed include statistics such as the number of records sent to Adabas Review, the number of active reports, the CPU-time used by Adabas Review and the number of I/Os performed by Adabas Review.

8

Getting Started

■ Accessing Adabas Review Online	68
■ Using Function Codes	71
■ Using Adabas Review Commands	71
■ Using the Online Help System	72
■ Accessing Adabas Online System (AOS) from SYSREVDDB	73
■ Accessing Adabas Review Repositories	73
■ Accessing Technical System Information	74
■ Ending a Session	80

This chapter tells you how to log on to Adabas Review online (SYSREVDDB), use the online help system, use function codes, PF keys and commands, and how to end your Adabas Review session.

Accessing Adabas Review Online

Adabas Review can be accessed online via a standard Natural application that resides in the Natural library SYSREVDDB.

➤ To access Adabas Review online via SYSREVDDB:

- 1 Access Natural as you do normally.
- 2 At the NEXT prompt, enter LOGON SYSREVDDB.
- 3 At the prompt, enter MENU.

If you are running in hub mode and if the default hub ID is specified as "AUTO", the value is determined from the SVC of the current Natural session. If there is a single hub running under the current SVC, that hub ID is used. If more than one hub ID is active, the following pop-up dialog appears. (If no hub is running, the value is determined as if "LFILE" were specified.)

```

NEXT menu                                     LIB=SYSREVDDB
Logon accepted to library SYSREVDDB.

                                     Available Review Hubs on SVC 237

                                     Please choose a Review Hub
-----
- 00041      _____
- 00558      _____
      _____
      _____
      _____
      _____
      _____
      _____
      _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---
                                Exit

```

Type any character next to the hub you want to use and press Enter.

Your Adabas Review administrator can tell you if an alternative logon procedure has been defined.

When you log on to Adabas Review, the Main Menu screen is displayed.



Note: When user exit P-UEXIT1 needs to be re-cataloged because of a new GDA, it is assumed, that the display programs need to be re-cataloged too, since they also use the SYS-REVDB GDA. In this case, a `RECAT ALL` will be triggered. See the `RECAT` command in the section *Command Reference*, in the *Adabas Review Reference Guide* for more information.

This section covers the following topics:

- [Main Menu](#)
- [General Use PF Keys](#)

Main Menu

13:04:49	A D A B A S - R E V I E W	2017-09-28
	Main Menu	Target: 00296
Code	Description	
AA	Available ADABAS Nuclei	
AH	Available Review Hubs	
AO	ADABAS Online System	
ER	Edit Report Definition	
ES	Edit Specialty Reports	
ET	Edit Target Definition	
LC	List Scheduled Reports	
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	Exit	Fin



Notes:

1. For an explanation of the commands in the **Code** column in the screen above, read *Commands* in the *Adabas Review Reference Guide*.
2. In the top right corner of most screens, Target: `nnnnn` will appear. This denotes the database or Hub where Adabas Review is running.

3. The Target may be changed to another target that is running Adabas Review by entering the DBID or HUBID directly into this field. You may also use the command line directives DBID=nnnnn or HUB=nnnnn to accomplish the same results.

General Use PF Keys

The following PF keys are available on many of the Adabas Review screens:

PF	Command	Description
1	Help	Access the help system.
3	Exit	Return to the previous screen. When this function key is pressed on the Main Menu, the Adabas Review Natural user exit, P-UEXIT3, is run.
7	-	Scroll backward through a list display.
8	+	Scroll forward through a list display.
10	<===	On VW/VX screens, this function key scrolls the report to the left. On other screens, this function key may have other meanings. Note: If you are using an external application, such as a session manager like Software AG's NETPASS, that assigns PF10 to a function of the external application, you may have problems viewing report results. If this is the case, you can use PF22 instead to scroll the report results to the left.
11	===>	On VW/VX screens, this function key scrolls the report to the right. On other screens, this function key may have other meanings. Note: If you are using an external application, such as a session manager like Software AG's NETPASS, that assigns PF11 to a function of the external application, you may have problems viewing report results. If this is the case, you can use PF23 instead to scroll the report results to the right.
12	Menu	Return to the Adabas Review Main Menu. When this function key is pressed on the Main Menu, the Adabas Review Natural user exit, P-UEXIT2, is run.
13	LR	The List Report Definition (LR) function lists all reports.
14	LS	The List Started Reports (LS) function lists reports that have been started.

Other Adabas Review commands and PF keys are local to a particular function or screen. These commands are usually shown on the relevant screen and have PF keys assigned to them.

Commands used in the list functions are not displayed on the screen. These commands may be displayed by using the online help system.

Using Function Codes

Adabas Review is menu-driven. A series of function codes and commands are used to navigate through its functions. Functions may contain several "layers" of screens.

Entering function codes singly accesses one layer at a time; entering a string of function codes enables you to bypass intervening screens and go directly to the screen you want.

For example, you can list the reports that have been started (LS) for Adabas Review by entering the following on the command line of the Adabas Review main menu:

```
LS
```

The LS code accesses the Started Reports screen.

Using Adabas Review Commands

Commands are used to perform specific tasks within a function.

➤ To issue an Adabas Review command:

- type the command on the command line and press ENTER

Or:

press the PF key corresponding to the command, if one has been defined.

A command may be included in a string, provided the command is the last element.

Some commands are standard throughout Adabas Review, and may be issued from any screen. These include the following:

Command	PF	Description
COLOR OFF	—	Returns to non-color display.
COLOR ON	—	Displays color attributes, if applicable.
EXIT	3	Terminates the function and returns you to the menu screen that precedes it. Note: When this command is entered on the Main Menu, the Adabas Review Natural user exit, P-UEXIT3, is run.

Command	PF	Description
FIN or QUIT	12	Terminates the Adabas Review session. Assigned to PF12 on the Adabas Review Main Menu. Note: When either of these commands are entered on the Main Menu, the Adabas Review Natural user exit, P-UEXIT2, is run.
HELP or ?	1	Displays the help screens for a particular screen or field. ? is used to display help for a specific field.
LOGO	—	Displays the Adabas Review logo screen.
MENU	12	Terminates the function, and returns you to the Adabas Review Main Menu.
MSG	—	Displays detailed explanations of Adabas Review messages. May be issued with or without a message number. If no message number is included, Adabas Review displays the explanation for the last message received, unless you have changed the Adabas Review screen or performed a different Adabas Review function since the message was displayed. In this case, specifying the MSG command without a message number produces an error.

Other Adabas Review commands are local to a particular system or function. These commands are usually listed on the relevant screen and have PF keys assigned to them.

Line commands used in the list functions are not displayed on the screen. These commands may be displayed by using the online help system.

Using the Online Help System

Online help is available for Adabas Review functions at any time during the Adabas Review session. You may view general help screens describing functions and commands or, if available, help windows for a specific data entry field:

➤ To access the general help screens:

- Press PF1

Or:

Enter the `HELP` command on the command line.

➤ To access specific help windows:

- Enter a question mark (?) on the data entry field.

If specific help is not available, the general help for the screen is displayed.

The following PF keys are available within the online help system:

PF	Command	Description
3	Exit	Exits from the display.
7	-	Scrolls one page backwards.
8	+	Scrolls one page forward.
9	-	Scrolls to the top of the text.
10	++	Scrolls to the bottom of the text.
12	Menu	Returns to the Adabas Review Main Menu.

Accessing Adabas Online System (AOS) from SYSREVDB

Adabas Online System (AOS) is a selectable unit of Adabas that enables database administrators to monitor and change aspects of an Adabas database interactively. Refer to the *Adabas DBA Tasks Manual* and the Adabas Online System documentation for more information.

If Adabas Online System is installed on your system and you have access privileges to it, you can access it from Adabas Review.

➤ **To access Adabas Online System from Adabas Review:**

- Enter the code `A0` on the command line.

Accessing Adabas Review Repositories

The Adabas Review repository is an Adabas file used for storing report definitions, historical data, and target definitions for Adabas Review.

Depending on the configuration at your site, more than one Adabas Review repository may be associated with your system. For example, if your site has Adabas Review installed on more than one database, there may be an Adabas Review repository for each database.

Using the SETFILE Command

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

The `SETALL` command differs from the `SETFILE` command in that it allows you to also change the current `DBID/HUBID` where Adabas Review is communicating. Changing this value has the same effect as entering `DBID=nnnnn` or `HUB=nnnnn` into the command line.

➤ **To access different Adabas Review repositories:**

- 1 Enter the command `SETFILE` or `SET` on the command line of any Adabas Review screen and press `ENTER`.
- 2 Provide the DBID and FNR for the Adabas Review repository you wish to access and press `ENTER`.

You are now able to access reports stored on a different Adabas Review repository. The setting remains until you either change it again or log off Adabas Review.

Accessing Technical System Information

The `TECH` command provides access to information about the Adabas Review system. It invokes the Database System Environment screen, which displays Adabas Review environmental and maintenance information. This function is useful for determining:

- the environment in which Adabas Review is executing; and
- the maintenance (zaps) applied to the current version of Adabas Review.

You can access the Database System Environment screen from:

- the Main Menu
- the List Report Definitions (LR) screen
- the List Started Reports (LS) screen
- the List Scheduled Reports (LC) screen

This section describes the information you can obtain:

- [Reviewing Technical System Information](#)
- [Reviewing Zap Information](#)
- [Reviewing Natural Object Zap Information](#)

Reviewing Technical System Information

➤ **To access technical system information:**

- Type `TECH` on the command line of one of the appropriate screens and press `ENTER`.

The `TECH` command Database System Environment screen is the first to appear:

```

13:07:29                A D A B A S   -   R E V I E W                2017-09-28
                        DATABASE System Environment                Target: 00296

Review Target
DBID/HUBID ..... 296
SVC ..... 249
Vers ..... 4.9_SP1

Review Repository
DBID ..... 20000
File ..... 149
Status ..... File Available

Natural
Natural Vers ..... 8.2_SP7
SYSREVDB Vers ..... 4.9_SP1
ADATMZ Vers ..... 8.3_SP4

TP Monitor
Monitor ..... COMPLETE
Vers ..... 6.8_SP1

REVEXIT1
Vers ..... 4.8_SP2
Status .... Installed Correctly
State ..... Active

REVEXIT2
Vers ..... 4.8_SP2
Status .... Installed Correctly
State ..... Not Active
Client ..... Off

Press PF2 for REVIEW fix list, ENTER to exit
Command: _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Fixes Exit  NatFx                                Menu

```



Note: The SVC that is displayed on the TECH screen is always the default SVC specified in the ADALNK routine. In addition, SVC information does not appear on this screen in BS2000 environments; BS2000 does not use SVCs.

From this screen, you can:

- Press PF1 to invoke help from any screen.
- Press PF2 to obtain information about updates (zaps) to Adabas Review nucleus code. The Review Fixes Information screen appears. For more information about these screens, read [Reviewing Zap Information](#), later in this section.
- Press PF4 to obtain information about updates (zaps) to Adabas Review database Natural objects. For more information, read [Reviewing Natural Object Zap Information](#), later in this section.
- Press ENTER to exit the Database System Environment screen.

The Database System Environment screen displays information about the following environment variables:

Data Category	Field	Description
Review Target	DBID/HUBID	The database ID in local mode or the hub ID in hub mode.
	SVC	The SVC of this Natural session. This is the SVC where SYSREVDDB is running. Note: The SVC that is displayed on the TECH screen is always the default SVC specified in the ADALNK routine. In addition, SVC information does not appear on this screen in BS2000 environments; BS2000 does not use SVCs.
	Vers	The version of Adabas Review, either local or for the given Adabas Review hub.
Review Repository	DBID	The database of the Adabas Review repository. The default corresponds to NTFIL241 in the Natural NATPARM parameter module. This database cannot be in the hub address space.
	File	The file number of the Adabas Review repository.
	Status	The status of the REVEXIT1 link routine exit for the TP monitor currently being used.
Natural	Natural Vers	The version of Natural currently installed at the user's site.
	SYSREVDDB Vers	The version of the Natural code of the Adabas Review software.
	ADATMZ Vers	The version of ADATMZ code used and loaded by Natural. ADATMZ is used by Adabas Review to obtain hub location information and to support the use of "AUTO" for the default Adabas Review target DBID or hub ID name in user profiles.
TP Monitor	Monitor	The type of TP system currently being used.
	Vers	The version of the TP monitor currently being used.
REVEXIT1	Vers	The version of the REVEXIT1 link routine exit for the TP monitor currently being used. REVEXIT1 is the new name for REVEXITB; REVEXITB is currently supported, but will no longer be supported in a future release.
	Status	The status of the repository identified by the repository DBID.
	State	The current state of REVEXIT1. Possible values are "Unknown", "N/A", "Active" and "Not Active" or "COR-controlled".
REVEXIT2	Vers	The version of the REVEXIT2 link routine exit for the TP monitor currently being used.
	Status	The status of the REVEXIT2 link routine exit for the TP monitor currently being used.
	State	The current state of REVEXIT2. Possible values are "Unknown", "N/A", "Active", "Not Active" or "COR-controlled".
	Client	The status of client reporting for Adabas Review. Valid values are "On", "Off", "Not Installed" or "n/a".

Reviewing Zap Information

➤ To obtain information about updates (zaps) that have been applied:

- Press PF2 from the Database System Environment screen.

The Review Fixes Information screen appears displaying maintenance (zap) information for the nucleus first:

[illegible]

From this screen, you can:

- Press PF1 to invoke help from any screen.
- Press PF2 to continue to cycle through the maintenance information screens, providing information on nucleus zaps, optional (special) nucleus zaps, REVEXIT1 zaps and optional (special) zaps, and zaps (updates) to Adabas Review Natural objects.
- Press ENTER to exit the Review Fixes Information screen.

Maintenance information screens display, by number, all of the zaps that have been applied to the current version of Adabas Review. When you press PF2, the maintenance screens cycle in the following sequence:

Screen Sequence	Review Zap Screen Title	Empower Naming Convention	Display Format on the TECH Screen
1	Zaps applied to the Review nucleus for target <i>nnn</i>	RD <i>vrsmnn</i>	<i>smnn</i>
2	Optional zaps applied to the Review Nucleus for target <i>nnn</i>*	RY <i>vrsmn</i> *	<i>smnn</i>
3	Zaps applied to the Link Routine (dependent) (for example, TSO ADALNK, CICS ADALINK)	AB <i>vrsmn</i> (BS2000 zaps) AD <i>vrsmnn</i> (Com-plete and TSO/batch zaps for z/VSE environments) AO <i>vrsmnn</i> (Com-plete and TSO/batch zaps for z/OS environments) CI <i>vrsmnn</i> (CICS zaps for all environments) II <i>vrsmnn</i> (IMS zaps for z/OS environments)	<i>smnn</i>
4	Zaps applied to the Link Routine (independent)	AI <i>vrsmn</i>	<i>smnn</i>
5	Zaps applied to REVEXIT1 and REVEXIT2	RD <i>vrsmnn</i>	<i>smnn</i>
6	Optional zaps applied to REVEXIT1 and REVEXIT2*	RY <i>vrsmnn</i> *	<i>smnn</i>
7	Updates to Review Natural objects	REV <i>vrSIsmn</i> REV <i>vrSX0nn</i>	IX <i>vrSi</i> <i>nn</i> where <i>i</i> is value I for fix 1 to 9, J for fix 10 to 19 and so on. IX <i>vrsmn</i>

* Optional zaps are not available in Empower.

Screen Sequence	Review Zap Screen Title	Empower Naming Convention	Display Format on the TECH Screen
Note: For link routine dependent and independent corrections, the <i>vrS</i> refers to the Adabas version (<i>v</i>), release (<i>r</i>), and maintenance level (<i>s</i>). In all other cases, it refers to the Adabas Review version (<i>v</i>), release (<i>r</i>), and maintenance level (<i>s</i>).			

The zap numbers are displayed in rows, from left to right in the order they were applied. If the maintenance screen displays released/test zaps, the test zaps are highlighted. Other maintenance screens display optional zaps.

The types of maintenance screens available vary with the operating environment in which Adabas Review resides. Any or all of the following maintenance information screens may be displayed:

- zaps (released/test, and optional) applied to Adabas Review nucleus for the database or hub;
- zaps (released/test, and optional) applied to Adabas Review link routine exit;
- INPLs (released/test) applied to the Adabas Review Natural code; and
- dependent and independent zaps applied to the Adabas link routine (ADALNK).



Note: For z/VSE systems, the first and the second screen display all zaps that are applied to the Adabas Review nucleus and the Adabas Review link routine exit.

Reviewing Natural Object Zap Information

➤ **To obtain information about updates (zaps) to Adabas Review database Natural objects:**

- Press PF4 from the Database System Environment screen.

The Review Fixes Information - Updates to Review Natural objects screen appears displaying the first set of maintenance (zap) information. From this screen, you can:

- Press PF1 to invoke HELP from any Technical Information function screen.
- Press ENTER to exit the Review Technical Information screen.

Ending a Session

An Adabas Review session may be ended in any one of the following ways:

➤ **To end the current function and return to the menu from which the function was called:**

- Press PF3; or enter the `EXIT` command on the command line and press ENTER .

➤ **To end the function you are using and return to the Adabas Review Main Menu:**

- Press PF12; or enter the `MENU` command on the command line and press ENTER.

➤ **To end your Adabas Review session from the Adabas Review Main Menu:**

- Press PF3 (Exit); or enter the `QUIT` command on the command line and press ENTER.

Or:

Press PF12 (Fin); or enter the `FIN` command on the command line and press ENTER.

➤ **To end your Adabas Review session without returning to the Main Menu:**

- Enter the `FIN` command on the command line and press ENTER

Or:

Enter the `QUIT` command on the command line and press ENTER.

9

Cost Accounting Example

■ Viewing Data Online	84
■ Processing Data in Batch Mode	85
■ Adabas Review Command Response Time Calculation	86

Adabas Review may be used to collect Adabas user resource consumption data that can be manipulated for cost accounting and charge-back purposes. This data can be processed and viewed online or in batch, or it can be downloaded to a personal computer for further processing.

This chapter describes the Adabas Review Cost Accounting Example.

Viewing Data Online

A sample cost accounting report "Cost Accounting Example" is supplied with Adabas Review. It may be used without modification, customized to suit site requirements, or used as a basis for other cost accounting reports. The report definition for the "Cost Accounting Example" report is shown below:

```

10:49:13                                A D A B A S   -   R E V I E W                                2016-05-21
                                           Edit Report                                           Target=15690

Detail/Summary: S
Report Name: COST ACCOUNTING EXAMPLE_____ DBID to Monitor: _____

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Field      | Order  | Sum   | Min  | Max  | Avg   | Pct   | Rate  | Cost  | Round |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| CQJOB_____ | 10     | -     | -    | -    | -     | -     | -     | -     | -     |
| TPUSERID_____ | 20     | -     | -    | -    | -     | -     | -     | -     | -     |
| IOS_____ |      | X     | -    | -    | -     | -     | -     | -     | -     |
| COMMANDS_____ |      | X     | -    | -    | -     | -     | -     | -     | -     |
| CMDRESP_____ |      | X     | -    | -    | -     | -     | -     | -     | -     |
| ADADURA_____ |      | X     | -    | -    | -     | -     | -     | -     | -     |
| _____ |      | -     | -    | -    | -     | -     | -     | -     | -     |
| _____ |      | -     | -    | -    | -     | -     | -     | -     | -     |
| _____ |      | -     | -    | -    | -     | -     | -     | -     | -     |
| _____ |      | -     | -    | -    | -     | -     | -     | -     | -     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
                                           Page 1
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

Command: _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---
      Help  Optns Exit  SOnly Save  Start  -      +      Sched Rules Flds  Menu  ␣

```

In this report, the generated NATURAL display program was modified to attach a cost value to the amounts of CPU time and Adabas time used, and to the number of Adabas calls and I/Os performed by each user.

By issuing the VW command, the report may be used to examine data that has been stored in the Adabas Review repository.

The following screen shows the type of information collected by the report:

21:15:59		COST ACCOUNTING EXAMPLE			2016-06-18
20:15:31		2016-06-18	Thru 21:15:17	2016-06-18	Target=15690
					Page: 1
CQ-Job	TPUserid	Total Num-of-IOs	Total Commands	Total Cmd-Resp	Total ADA-Dur
ALSCCOMP		210	151	1.1202	5.152688
*****	SUB-T==>	210	151	1.1202	5.152688
COMPLETE		6	12	0.0702	0.251056
	USER1	318	652	1.3057	6.523424
	USER2	57	186	0.2409	0.903504
	USER3	1241	2775	5.5287	26.014992
	USER4	97	219	0.3831	2.337776
	USER5	271	831	1.2486	7.082896
	USER6	169	147	0.4203	2.965808
	USER7	117	129	0.4519	2.433072
	USER8	136	386	0.5802	3.022160
	USER9	5573	8030	7.7051	107.307856
	USER10	323	707	1.3217	6.409696
Command: _____					
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---					
Help Sort Exit Menu					

Additionally, the displayed resource consumption data may be downloaded to a personal computer using the DL command. To use this feature, the Software AG product Entire Connection must be installed.

For more information, read *Downloading Report Output* in the *Adabas Review User's Guide*.

Processing Data in Batch Mode

Cost accounting values can be defined for batch command log processing by using the COST batch parameter statement.

The member REVCOST, shown below, is a sample report for cost accounting that uses the COST statement. REVCOST is located in the Adabas Review source data set.

```
*
* This sample report allocates a cost value to three summary
* fields based on the JOBNAME and Adabas user's TPUSERID.
* In this example, each I/O is assigned a cost of $0.02,
* each command response second is valued at $0.25 and each second of
* Adabas duration costs $0.10.
*
REPORT TYPE=SUMMARY,
      TITLE='COST ACCOUNTING EXAMPLE',
```

```
PROGRAM=SR-00028,  
RESTART=Y,  
MAXSTORE=8  
DISPLAY JOBNAME,TPUSERID  
SUM      IOS,CMDRESP,ADADURA  
COST      IOS*0.02,CMDRESP*0.25,ADADURA*0.10  
*
```

For additional information about Adabas Review batch processor parameter statements, read *Using Batch Facilities*, in the *Adabas Review User's Guide*.

A sample of the data produced by the batch report is shown on the next section.

Adabas Review Command Response Time Calculation

The Adabas Review command response time calculation is based on the following information.

Adabas Review stores in the command table the minimum Adabas duration for each command type returning a zero response code.

The command table is updated whenever a lower duration value is encountered.

Index

Symbols

+ command, 70, 73
++ command, 73
- command, 70, 73
-- command, 73
? command, 72

A

Adabas
 interregion communication, 9
 multiple nuclei, 5
 user buffer extension, 13
Adabas Cluster Services, 5, 10
Adabas Online System, 6
 accessing, 73
Adabas Online System (AOS)
 AO function, 73
Adabas Parallel Services, 5, 10
Adabas Review
 accessing, 68
 allocating storage for, 27
 defined, 5
 purpose of, 5
 structure of, 5, 10, 12
 summary of features, 5
 supplied reports, 6
Adabas Review DB
 nucleus and subsystems, 10
Adabas Review reports
 about, 33
ADACLX
 command log extension module, 11
ADALNK, 9
ADALOG
 command logging module, 11
 interface client component, 13
ADAMPM, 9
ADAREV
 hub server component, 10
ADARUN parameters
 NAB to size the attached buffer, 28-29
 NC to size the command queue, 28
 REVIEW, 13
ADARVU
 interface client component, 11, 13
 transferring data to the hub, 11
ADASVC, 9

AO function
 Adabas Online System (AOS), 73
AOS (see Adabas Online System (AOS))
attached buffer, 29
autostarted reports, 43

B

batch data processing, 17
BUFFER-SEGMENTS parameter, 30

C

client reporting
 about, 47
 ADARUN REVIEW parameter setting considerations, 52
 data collected, 51
 installation and activation, 48
 process, 49
 reporting on client reporting data, 52
client reporting data, 5
client/server approach, 5
client/server architecture, 9
 diagram, 12
CMDRESP field, 86
COLOR command, 71
command features
 scrolling, 73
command log file
 description, 22
command log processing, 5
command queue, 28
command response time
 calculation, 86
commands
 issuing, 71
 overview, 71
 standard, 71
 types used with reports, 34
 used in list functions, 72
 used to access Adabas Review information, 74
 used to access system information, 74
components
 hub server, 10, 12
 interface client, 11-12
cost accounting
 example, 84
 processing data in batch mode, 85
 viewing data online, 84

D

- data types supported, 6
- database
 - statistics functions, 73
- database fields
 - user-defined, 6
- Database System Environment screen, 74
- databases
 - multiple, 5
- default user profile, 26
- Display programs
 - using Software AG Editor, 55

E

- ending a session, 80
- EXIT command, 71, 73, 80
- exits, 16

F

- FIN command, 72, 81
- fix information, 77
- function codes
 - entering a string of, 71
 - overview, 71

G

- GENAUTO command, 43
- GENCARD command, 43

H

- HELP command, 70, 72
- help system
 - accessing, 72
 - online, 70, 72
 - PF keys, 72
- history data, 5
 - collecting, 15
- hub
 - requirements for use, 9
- hub concept, 5
- hub mode
 - collecting data in, 14
 - command log processing, 13
- hub server
 - advantages of, 10
 - attached buffer storage requirements, 29
 - command queue storage requirements, 28
 - components, 10, 12
 - defined, 10
 - location, 9
 - storage requirements, 27-28
 - transferring data to, 11

I

- interface client
 - components, 11-12
 - defined, 11

- location, 9
- transferring data to the hub, 11

L

- link routine exits, 13
- list functions
 - for reports, 44
- local mode
 - collecting data in, 8
 - interactive and batch execution modes, 8
 - storage requirements, 32
- logging off, 80
- logging on, 68
- LOGO command, 72
- LR command, 70
- LS command, 70

M

- main menu, 69
- MAXSTORE
 - report parameter, 31
- MENU command, 70, 72-73, 81
- MSG command, 72
- multiple databases
 - monitoring, 14

O

- online data processing, 15

P

- PF keys
 - general, 70
- processing data
 - in batch, 17
 - online, 15

Q

- QUIT command, 72, 81

R

- RAOSAUTO, 43
 - hub server component, 10
- RAOSHIST, 15
 - hub server component, 10
- raw log files
 - description, 24
- reports
 - about, 33
 - autostarted, 43
 - batch mode, 41
 - commands used with, 34
 - components, 34
 - database fields, 35
 - display programs, 35, 40
 - interactive mode, 41
 - list functions, 44
 - MAXSTORE parameter, 31

- option parameters, 35, 38
- overview, 34
- processing rules, 35, 37
- report definition components, 35
- report definitions, 34-35
- scheduling, 41
- starting, 41
 - batch mode, 41
 - interactive mode, 41
- stopping, 41
- storage for control blocks, 31
- storage for data collection areas, 31
- storage requirements, 27, 31
- supplied, 35
- total storage limit (MAXSTORE), 31
- user exits, 39
- repositories, 8, 13
 - accessing, 74
 - defined, 19, 73
 - multiple, 19, 73
 - restricted location in hub mode, 19
- REVEEXIT1 description, 16
- REVEEXIT2 description, 16
- REVEEXITB description, 16
- REVHUB
 - hub server component, 10
- Review Fixes Information screen, 77
- Review Technical Information screen, 80
- REVIEW-BUFFER, 13
 - storage requirements, 27, 30
- REVIEWB
 - hub server component, 13
- RVUALT
 - alternate history data, 15
- RVUAUT1, 43
- RVUAUT2, 43
- RVUPRT00 output printer
 - description, 19
- RVUPRTnn output printer
 - description, 20

S

- scroll commands, 73
- session
 - ending, 80
- SET command, 19, 73
- SETA command, 19
- SETFILE command, 19, 73
- Storage
 - allocating above the 16MB line, 27
 - allocating from MVS subpool 5, 27
- storage
 - MAXSTORE report parameter, 31
 - requirements, 27
- summary log file
 - description, 23
- summary reports
 - storage requirements for sorting, 31
- supplied reports, 35
- SYSREVDDB
 - accessing online, 68
 - description, 15
- system information

- accessing, 74

T

- TECH command, 74
- technical information
 - Adabas Review database Natural zaps, 80
- terminating a session, 80
- Trans-port, 9

U

- user exits
 - control transfer of data to the hub, 11
 - report, 39
- user profile system
 - administrator profiles, 26
 - default profile, 26
 - default user profile, 26
 - group profile, 26
 - maintaining user profiles, 26
 - purpose, 26
- user profiles
 - default, 26
- users
 - storage requirements for online use, 27, 32

V

- VIEW command, 40

W

- work areas
 - storage requirements, 27, 32

Z

- zaps
 - list of applied Adabas Review database Natural, 80
 - list of applied zaps, 77
 - list of optional, 80

