

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 OS/390 and z/OS environments, Adabas Review allocates all storage from MVS subpool 5. This allows you to accurately determine the exact amount of storage Adabas Review is using with an MVS monitoring package.

This chapter covers the following topics:

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

Storage for the Hub

Review 4.3 uses the high performance Trans-Port protocol as its primary method of sending data from the ADABAS nucleus to the Review Hub. When running in this mode, the NC and NAB ADARUN parameters mentioned in the paragraphs below are not particularly meaningful. The storage that NC and NAB control is used for Review administrative traffic only, and the workload from the ADABAS nuclei does not require the storage managed by these parameters. These two parameters can be left at the default values if all nuclei are using Trans-Port.

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

Two queues, both controlled by the database administrator (DBA), are used by the Adabas Review hub: the command queue (sized using the ADARUN parameter NC) and the attached buffer (sized using the ADARUN parameter NAB). See the *Adabas Operations Documentation* for more information.

Command Queue

The command queue stores information about the client nucleus such as job name, internal ID, etc. Each entry in the command queue represents one command log record from a client if Trans-Port is not being used.

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. The command queue is also used for administrative calls from the Review user interface.

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 command per second, then the expected arrival rate at the hub is also 2000 commands log records per second.

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

Attached Buffer

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

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

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

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

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

Example

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

One approach would be to compute

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

where 4096 is the size of one NAB segment. If NC=1000 (see the example in section *Command Queue*), the starting value would be

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

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

Storage for the REVIEW-BUFFER

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

The BUFFER-SEGMENTS parameter specifies the size of the REVIEW-BUFFER. Each buffer segment is 128 bytes. The default value for BUFFER-SEGMENTS is as follows:

Operating System	Default Value
OS/390 or z/OS	2000 (250 kilobytes)
VSE/ESA	700 (87 kilobytes)
z/VM or BS2000	400 (50 kilobytes)

For OS/390, z/OS, VSE/ESA, and BS2000

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

For z/VM

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

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

- A *smaller* REVIEW-BUFFER means that REVIEWB is called more often but with fewer command log records to process so that the processing delay is less for each processing cycle.

Note:

The Adabas RVUEXI parameter CMS-FULLSYNCH may be used to force REVIEWB to process each command log instead of waiting for the REVIEW-BUFFER to become full.

Storage for Reports

For Control Blocks

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

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

For Data Collection Areas

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

Total Storage Limit

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

Storage for Online Users

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

- Each request from the Adabas Review online system results in the allocation of a piece of storage (about 500 bytes or $\frac{1}{2}$ kilobyte) that is deallocated when the request has been satisfied.
- To maintain the integrity of each request, Adabas Review allocates an area for each user requesting Adabas Review to service a request. For example, a request to view a report requires an Adabas call for each record that is to be viewed online.
- In cases where more than one user is viewing the same or different reports, Adabas Review must remember the status of each user between Adabas calls.

Storage for Work Areas

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