

# Useful ADARUN Parameters for Adabas Parallel Services

This chapter provides a reference for the ADARUN parameters useful to Adabas Parallel Services processing. For information on how to specify ADARUN parameters, read the *Adabas Operations Manual* documentation provided with Adabas.

- CLOGMRG Parameter: Cluster Command Log Merge
  - CLUCACHEEXTRA Parameter: Adjust Shared Cache Vector Size
  - CLUCACHESIZE Parameter: Size of the Global Cache Area
  - CLUCACHETYPE Parameter: Global Cache Area Storage Type Option
  - CLUCACHEUNCHANGED Parameter: Global Cache Unchanged Block Option
  - CLUGROUPNAME Parameter: Cluster Group Name
  - CLULOCKSIZE Parameter: Size of the Global Lock Area
  - CLUSTER Parameter: Adabas Nucleus Cluster Session Control
  - DIRRATIO / ELEMENTRATIO Parameters: Cluster Cache Directory-to-Data Elements Ratio
  - LDEUQP Parameter: Length of Unique Descriptor Pool
  - LRDP Parameter: Length of the Redo Pool
  - MLWTO Parameter: Multi-Line Console Message Control
  - MXCANCEL Parameter: Timeout Threshold for Canceled Peer Nucleus
  - MXCANCELWARN Parameter: Timeout Threshold for Canceled Peer Nucleus Warning
  - MXMSG Parameter: Timeout Threshold for Internucleus Command Processing
  - MXMSGWARN Parameter: Timeout Threshold for Internucleus Command Processing Warning
  - MXWTOR Parameter: Self-Termination Operator Query Interval
  - NUCID Parameter: Adabas Cluster Nucleus ID
  - PROGRAM Parameter: Program to Run
  - SMFDETAIL Parameter: Selecting Detail SMF Record Sections
  - V64BIT Parameter: 64-Bit Virtual Storage Option
-

## CLOGMRG Parameter: Cluster Command Log Merge

Parameter	Specify . . .	Possible Values	Default
CLOGMRG	whether the Adabas cluster is to run with automatic CLOG merge.	YES   NO	NO

An automatic CLOG merge process provides a single cluster-wide command log containing, in chronological order, all Adabas commands executed by any of the cluster nuclei in the time period covered by the log.

CLOGMRG specifies whether the Adabas cluster is to run with the automatic CLOG merge:

Value	Action
YES	enables an automated process that copies dual or multiple command logs to sequential data sets and merges those data sets across a cluster into a single cluster-wide command log. This process automatically merges command logs when an ADARES CLCOPY is submitted from user exit 2 and executed. The cluster nuclei and the ADARES utility perform coordinated CLOG switches and CLCOPY functions with merging of CLOG records, much the same way the PLOG merge process works.
NO	(the default), each nucleus produces its own CLOG, independent of any other nucleus in the cluster; dual or multiple CLOGs can be manually copied using ADARES CLCOPY to a sequential data set; and the resulting sequential CLOGs can be manually merged across a cluster using the ADARES MERGE CLOG function.

CLOGMRG is a global (that is, a cluster-wide) parameter, which means that the setting of this parameter in the first cluster nucleus to become active is propagated to all nuclei that subsequently become active.

### Parameter Dependencies

It is possible to specify both LOGGING=NO and CLOGMRG=YES. In a cluster environment, it is also possible that a CLOG will be written to even if LOGGING=NO. As long as CLOGMRG=NO, nothing will be written to the CLOG if LOGGING=NO. However, if LOGGING=NO and CLOGMRG=YES, control records necessary for ADARES CLCOPY will be written to the CLOG.

CLOGMRG is effective only in an Adabas cluster environment; that is, when CLUSTER is specified with a value other than NO and NUCID is specified with a nonzero value.

### Dynamic Modification

The setting of the CLOGMRG parameter can be changed dynamically using the CLOGMRG command from the operator console, the ADADBS OPERCOM CLOGMRG function, or the Modify Parameter function of Adabas Online System.

Because CLOGMRG is a global parameter, the change request to one nucleus is automatically propagated to all nuclei in the cluster.

## CLUCACHEEXTRA Parameter: Adjust Shared Cache Vector Size

Parameter	Specify . . .	Possible Values	Default
CLUCACHEEXTRA	the additional vector size, in blocks, required to allow Adabas Cluster Services and Adabas Parallel Services to track the blocks held by Adabas Caching Facility in the different cluster nuclei.	0 - 2147483647	1000

This parameter is local and unchangeable. It has effect only if both a cluster product (Adabas Cluster Services or Adabas Parallel Services) and Adabas Caching Facility are active.

If Adabas Cluster Services or Adabas Parallel Services are running on the same system as the Adabas Caching Facility, use the CLUCACHEEXTRA parameter to adjust the vector size of the shared cache. This ensures that Adabas Cluster Services and Adabas Parallel Services can track all of the blocks held privately by the Adabas Caching Facility.

Set the value to the maximum number of blocks expected to be cached by the Adabas Caching Facility. This number can be derived from the CASSOMAXS, CDATAMAXS, and CMAXCSPS parameter settings of the Adabas Caching Facility and the ASSO and DATA block sizes of the database.

## CLUCACHESIZE Parameter: Size of the Global Cache Area

Parameter	Specify . . .	Minimum	Maximum	Default
CLUCACHESIZE	the amount of storage to allocate for the global cache area that services the Adabas Parallel Services cluster in which the current nucleus participates.	128K	depends on the type of cache (CLUCACHETYPE parameter)	none

The CLUCACHESIZE parameter specifies the amount of storage to be allocated for the global cache area which all nuclei in the Adabas Parallel Services cluster use to share Associator and Data blocks from the database. It has an effect only if CLUSTER=LOCAL is also specified.

The location and maximum size of the global cache area depends on the CLUCACHETYPE parameter. If CLUCACHETYPE=DSP, the global cache is kept in a dataspace shared between the Adabas Parallel Services nuclei. Otherwise, the cache is kept in shared 64-bit virtual storage (z/OS systems only).

CLUCACHESIZE must be specified for Adabas Parallel Services nuclei; there is no default.

Each Adabas Parallel Services nucleus has its own buffer pool sized by the ADARUN parameter LBP . Optionally, it may also have one or more local cache spaces administered by the Adabas Caching Facility (refer to the description of the ADARUN parameter CACHE). In addition, Adabas Parallel Services uses a global cache area to ensure that current images of Associator and Data blocks from the database are available to all Adabas Parallel Services cluster nuclei.

The global cache area must be large enough to retain:

- directory elements for all blocks that reside in all buffer pools and the global cache itself; and
- enough data elements to keep changed blocks between buffer flushes (cast-outs).

The ADARUN parameters DIRRATIO and ELEMENTRATIO determine the ratio between the number of directory and data elements.

CLUCACHESIZE may be specified in bytes, in kilobytes followed by a "K", in megabytes followed by an "M", or in gigabytes followed by a "G":

- the minimum size is 131072 bytes (or 128K);
- for CLUCACHETYPE=DSP, the maximum size is 2147483147 bytes (or 2097152K; or 2048M; or 2G); and
- the maximum allowed numeral is 2147483147; for CLUCACHETYPE=V64, use the K, M or G suffix (for example, 10G or 10240M or 10485760K for a global cache of 10 gigabytes size).

The value is rounded up to the nearest 4-kilobyte boundary.

Appropriate sizes for the global cache area depend very much on the application load in the cluster. For more details about these calculations, read *Global Cache Area Size*. You can use the Adabas Online System cache structure size calculator described in *Estimating Sizes for the Cache Structure in a Cluster Environment* to determine appropriate global cache area sizes.

### Example

In the following example, the global cache area allocated is 500 M.

```
ADARUN PROG=ADANUC,CLUCACHESIZE=500M
```

## CLUCACHETYPE Parameter: Global Cache Area Storage Type Option

Parameter	Specify . . .	Valid Values	Default
CLUCACHETYPE	the virtual storage type for the global cache area	DSP   V64   L64	DSP

The CLUCACHETYPE parameter specifies the type of virtual storage to be used for the global cache of an Adabas Parallel Services cluster. It has an effect only if CLUSTER=LOCAL is also specified. Valid values are "DSP" (available on all supported operating systems) and "V64" or "L64" (available only on z/OS systems).

The size of the global cache is specified in the CLUCACHESIZE parameter.

The default value is "DSP", indicating that a shared dataspace will be used for the global cache. In this case, the maximum cache size is 2 GB.

If "V64" is specified (on z/OS systems), a shared storage area in 64-bit virtual storage will be used for the global cache. In this case, the cache can be made much larger than 2 GB, but is still subject to the MEMLIMIT value in effect for the ADACOM job for the Adabas Parallel Services database.

If "L64" is specified (on z/OS 1.9 or later systems running on IBM z10 machines), the global cache space shared by the cluster nuclei will reside in 64-bit virtual storage that is backed by large pages (provided the large page pool has been configured to a sufficient size and is available in the system).

#### Notes:

1. If one ADACOM job is used to hold the shared cache and lock spaces for multiple Adabas Parallel Services clusters, L64 cache spaces of different clusters are slightly less isolated from one another than DSP or V64 cache spaces, creating a marginally higher risk that a cache-related error in one cluster might induce an error in a different cluster. This very small risk can be avoided altogether by using different ADACOM jobs for different L64 cache spaces, but this is not mandatory.
2. Virtual 64-bit storage backed by large pages can only be used on IBM z10 mainframes running z/OS Version 1.9 or above and for which IBM large page support has been enabled via IBM APAR OA20902. You must also allocate the size of the large page pool (use the LFAREA parameter in the IEASYSxx member of SYS1.PARMLIB). The LFAREA parameter allows you to specify the amount of real storage to be used for large pages; this parameter cannot be changed dynamically and, if it is not set, page frames will remain allocated at a size of 4K.

#### Example

In the following example, the type of virtual storage used for the global cache area is specified as "V64".

```
ADARUN PROG=ADANUC,CLUCACHETYPE=V64
```

## CLUCACHEUNCHANGED Parameter: Global Cache Unchanged Block Option

Parameter	Specify . . .	Possible Values	Default
CLUCACHEUNCHANGED	whether unchanged blocks are written to global cache during Adabas Parallel Services processing.	YES   NO	NO

The CLUCACHEUNCHANGED parameter specifies whether an Adabas Parallel Services nucleus should write unchanged blocks to the global cache. It has an effect only if CLUSTER=LOCAL is also specified. Valid values are "YES" and "NO".

- The default value is "NO", which indicates that the nucleus will write only changed (updated) Associator and Data blocks from the database to the global cache.
- If "YES" is specified, the nucleus will write changed and unchanged Associator and Data blocks to the global cache.

### Recommendation

If a lot of virtual storage can be made available for local and global caching by the Adabas Parallel Services cluster (where local caching is controlled by the ADARUN LBP parameter and possibly the Adabas Caching Facility add-on product), assigning the bulk of the available storage to the global cache (ADARUN CLUCACHESIZE parameter), which is shared by all Adabas Parallel Services nuclei for the database, provides for more economic use of the storage than assigning it to the local caches of the individual cluster nuclei (ADARUN LBP parameter and the Adabas Caching Facility ADARUN CASSOMAXS and CDATAMAXS parameters), which are private to each nucleus. However, each cluster nucleus requires sufficient local buffer pool space (LBP parameter) to hold the entire working set of Associator and Data blocks needed for parallel command processing.

In cases where the global cache size (CLUCACHESIZE parameter) is set to a large value, CLUCACHEUNCHANGED should be set to "YES", so that the global cache achieves a better hit rate (efficiency).

### Example

In the following example, the CLUCACHEUNCHANGED parameter is set to that both changed and unchanged Associator and Data blocks are written to the global cache.

```
ADARUN PROG=ADANUC, CLUCACHEUNCHANGED=YES
```

## CLUGROUPNAME Parameter: Cluster Group Name

Parameter	Specify . . .	Possible Values	Default
CLUGROUPNAME	the name of the XCF messaging group in Adabas Cluster Services.	text	none
	the name of the messaging group in Adabas Parallel Services.	text	the router name

The CLUGROUPNAME parameter need not be specified for Adabas Parallel Services except when Adabas Parallel Services is used to recover an Adabas Cluster Services database after a failure that resulted in a remnant DIB entry. In this case, the CLUGROUPNAME parameter for Adabas Parallel Services must be set to the same value as it was for the failed Adabas Cluster Services nuclei.

For Adabas Cluster Services, the CLUGROUPNAME parameter provides the XCF group name to be used by the cluster nuclei for their communication using XCF.

All nuclei in the same cluster must have the same CLUGROUPNAME. Nuclei belonging to different clusters must have different CLUGROUPNAMEs.

In addition, the CLUGROUPNAME must be different from the XCF group names used by other subsystems that communicate using XCF.

The name can be 1-8 characters long. Valid characters are numeric, uppercase alphabetic, and the special characters \$, @, and #. The name must begin with an alphabetic character (A-Z). IBM names begin with SYS, UNDESIG, or the characters A through I (these characters are not reserved).

### Example

The following examples specifies the name of the XCF messaging group as "SAG1CLU".

```
ADARUN PROG=ADANUC,CLUGROUPNAME=SAG1CLU
```

## CLULOCKSIZE Parameter: Size of the Global Lock Area

Parameter	Specify . . .	Minimum	Maximum	Default
CLULOCKSIZE	the amount of storage allocated for the global lock area	128K	2G	none

The CLULOCKSIZE parameter specifies the amount of storage to be allocated for the global lock area which all nuclei in the Adabas Parallel Services cluster use to synchronize their processing. It has an effect only if CLUSTER=LOCAL is also specified.

CLULOCKSIZE must be specified for Adabas Parallel Services nuclei; there is no default.

CLULOCKSIZE may be specified in bytes, in kilobytes followed by a "K", in megabytes followed by an "M", or in gigabytes followed by a "G".

- The minimum size is 131072 bytes (or 128K)
- The maximum size is 2147483147 bytes (or 2097152K, 2048M, or 2G).

The value is rounded up to the nearest 4-kilobyte boundary.

Appropriate sizes for the global lock area depend on the expected amount of parallel command processing in the cluster. For more details about these calculations, read *Global Cache Area Size*. You can use the Adabas Online System lock structure size calculator described in *Estimating Sizes for the Lock Structure in a Cluster Environment* to determine appropriate global lock area sizes.

## CLUSTER Parameter: Adabas Nucleus Cluster Session Control

Parameter	Specify . . .	Possible Values	Default
CLUSTER	whether the Adabas nucleus will participate in a cluster of nuclei working on the same Adabas database, and, if so, the type of cluster (Adabas Parallel Services or Adabas Cluster Services) in which it will participate.	NO   SYSPLEX   LOCAL	NO

The CLUSTER parameter specifies whether the Adabas nucleus will participate in a cluster of nuclei working on the same Adabas database, and if so, which type of cluster:

- If CLUSTER=NO (the default), the nucleus will not participate in an Adabas cluster. If CLUSTER=NO, the NUCID parameter must be set (or default) to 0.
- If CLUSTER=LOCAL, the nucleus will participate in a cluster where all nuclei run on the same operating system image. The add-on product Adabas Parallel Services is required to run in this mode.
- If CLUSTER=SYSPLEX, the nucleus will participate in a cluster where the nuclei may run on different systems within a parallel sysplex. This is supported only for z/OS systems. The add-on product Adabas Cluster Services is required to run in this mode.

If CLUSTER=LOCAL or SYSPLEX, the NUCID parameter must be set to a nonzero value that identifies the individual cluster nucleus. Also, the parameter settings MODE=SINGLE, READONLY=YES, LFIOP=0 and DTP=TM, and the use of sequential protection log data sets (DD-name/link name DDSIBA) are not allowed for cluster nuclei.

### Example

The following example includes the nucleus in a cluster run with Adabas Cluster Services:

```
ADARUN DBID=53,CLUSTER=SYSPLEX,NUCID=5301
```



## DIRRATIO / ELEMENTRATIO Parameters: Cluster Cache Directory-to-Data Elements Ratio

The DIRRATIO and ELEMENTRATIO parameters together define the ratio of directory elements to data elements in the Adabas Cluster Services coupling facility cache structure or the Adabas Parallel Services global cache area.

This ratio influences the allocation of the cache structure/area when the first cluster nucleus starts. It has no influence on the cache structure/area when a subsequent nucleus starts.

Parameter	Specify . . .	Minimum	Maximum	Default
<u>DIRRATIO</u>	the directory portion of the directory-to-data element ratio	1	32767	4
<u>ELEMENTRATIO</u>	the element portion of the directory-to-data element ratio	1	32767	1

The DIRRATIO / ELEMENTRATIO ratio is optionally used in Adabas cluster environments to tune the cache structure/area for better performance.

Directory elements are used to keep track of the cluster members that have individual blocks in their local caches (buffer pool and Adabas Caching Facility caches), so that the blocks can be invalidated should other members modify them.

Data elements are used to hold the contents of Associator and Data blocks from the database.

If the number of directory elements is insufficient, existing directory elements are reused and the blocks associated with those directory elements are invalidated and can then no longer be reused. When blocks become invalid even though they have not been modified, they must be reread and registered again the next time they are referenced and validated.

It is generally better to reassign storage for data elements to keep more Associator and Data blocks in the cache structure/area than to define too many directory elements in the cache. More data elements than necessary can be used to keep additional blocks to improve the cache efficiency.

The number of directory elements need not be greater than the sum of the sizes of all buffer pools plus the amount of space represented by data elements (number of data elements times data element size), divided by the smallest Associator block size.

You can use the Adabas Online System cache structure size calculator described in *Estimating Sizes for the Cache Structure in a Cluster Environment* to determine appropriate global cache area sizes and DIRRATIO/ELEMENTRATIO parameter values.

When connecting to the cache structure/area during startup, the ADAX57 message reports the number of directory and data elements allocated and the size of a data element.

## LDEUQP Parameter: Length of Unique Descriptor Pool

Parameter	Specify . . .	Minimum	Maximum	Default
<u>LDEUQP</u>	the size of the unique descriptor pool.	5000	16777215	5000

### Note:

Zap AN822111 has increased the minimum setting size of the ADARUN LDEUQP parameter from zero to 5000 bytes.

The unique (UQ) descriptor pool comprises elements that are the length of a unique descriptor value plus 14 bytes. The UQ descriptor pool should be large enough to hold all unique descriptor values that are being changed by *all* transactions at any point in time.

An Adabas nucleus will make an entry in this pool every time a record with a unique descriptor is deleted or altered such that an existing descriptor value is *deleted* from the index. Adabas Cluster Services and Adabas Parallel Services nuclei also make an entry when a record with a unique descriptor is *inserted*. Entries in the unique descriptor pool generally exist for the duration of a transaction.

### Specific Product Recommendations

- Consider adjusting the value of this parameter when the SLOG file is defined for use on the Event Replicator Server, causing replication data to be written and read from the SLOG file.
- If the Event Replicator Server is set to support updates by multiple concurrent users to Adabas targets (when the NPADACALLS initialization parameter is set to any value greater than "1"), consider adjusting the value of this parameter in the target Adabas nucleus to ensure the target nucleus can handle updates from multiple concurrent users.

### Example

For example, assuming that the average length of unique descriptor values is about 16 bytes, the following Adabas session can process a maximum of about 1000 unique descriptors at one time.

```
ADARUN PROG=ADANUC,LDEUQP=30000
```

## LRDP Parameter: Length of the Redo Pool

Parameter	Specify . . .	Minimum	Maximum	Default
<u>LRDP</u>	the size (in bytes) of the redo pool.	0; or 80000	2147483647	<p>For Adabas Parallel Services 8.2 or later, the default is 0 (zero).</p> <p>For Adabas Cluster Services 8.2 or later, the default is either the value of the ADARUN LFIOP parameter or 10 megabytes (10M), whichever is smaller.</p> <p>For earlier releases of Adabas Parallel Services and Adabas Cluster Services, the default is the value of the ADARUN LFIOP parameter.</p>

The LRDP parameter specifies the size of the redo pool in an Adabas Cluster Services or Adabas Parallel Services nucleus. It has an effect only when CLUSTER=SYSPLEX or LOCAL has also been specified. For Adabas Parallel Services 8.2 or later, its default is zero (0). For Adabas Cluster Services 8.2 or later, its default is either 10 megabytes (10M) or the value of the ADARUN LFIOP parameter, whichever is smaller. For earlier releases of Adabas Parallel Services and Adabas Cluster Services, the default is the value of the ADARUN LFIOP parameter.

- If LRDP is set to zero, the cluster nucleus performs immediate publishing: Updated database blocks are always written to the global cache at the time each update takes place.
- If LRDP is set to (or defaults to) a nonzero value, the cluster nucleus performs deferred publishing: The writing of updated database blocks to the global cache is deferred until later. However, updates are always published in the global cache before the transactions they belong to are committed (by ET commands). The redo pool is used to maintain descriptions of all unpublished updates, in case they need to be redone due to concurrent updates to the same blocks by other nuclei in the cluster.

By combining multiple updates to the same block into a single write-to-cache operation, deferred publishing tends to result in fewer read and write operations from and to the global cache than immediate publishing. Depending on the type of application workload, this may have a large positive effect on performance. This is especially the case for Adabas Cluster Services, where frequent communication with the cache in the coupling facility may cause a high overhead.

**Recommendation:** For Adabas Cluster Services, specify a nonzero LRDP parameter (or use the default).

Different nuclei in the same cluster can have different settings of LRDP. It is also possible, although not recommended, to run one nucleus with LRDP=0 and another one with LRDP>0.

If the redo pool specification is too small and the pool runs full, the nucleus will write additional updated blocks to the global cache before the surrounding transactions end. Such additional cache writes may hurt system performance. To obtain more information about the current usage of the redo pool, review the redo high-water mark in the output of the DRES operator command and in the cluster nucleus shutdown statistics. This information is also available in corresponding Adabas Online System (AOS) displays and in the output of the ADADBS OPERCOM function.

## MLWTO Parameter: Multi-Line Console Message Control

### Note:

At this time, this parameter only affects the appearance of replication-related and cluster-related messages (messages with prefixes ADAF\*, ADAX\*, PLI\*, and PLX\*).

Parameter	Specify . . .	Possible Values	Default
MLWTO	Whether multi-line messages that are normally written to the operator console as a series of single-line write-to-operator (WTO) messages should instead appear as multi-line WTO messages.	YES   NO	NO

To activate multi-line console messages, set MLWTO to YES in the nucleus startup job. When MLWTO is set to NO (or is allowed to default), no multi-line console messages will appear.

Multi-line console messages are only supported in z/OS and z/VSE environments. Specifying MLWTO=YES in BS2000 or z/VM environments has no effect on the console messages.

### Example:

```
ADARUN PROG=ADANUC,MLWTO=YES
```

## MXCANCEL Parameter: Timeout Threshold for Canceled Peer Nucleus

Parameter	Specify . . .	Minimum	Maximum	Default
<u>MXCANCEL</u>	the timeout threshold, in seconds, in which the processing of an intracluster cancellation request must occur.	15	2147483647	MXMSG parameter value

The optional MXCANCEL parameter specifies the maximum number of seconds for which one cluster nucleus (the sending nucleus) waits for the termination of another nucleus (the peer nucleus) in the cluster that it has canceled for not responding to an internal, intracluster command. If a value for MXCANCEL is not specified, the value specified for the ADARUN MXMSG parameter is used. Values from zero through 14 are not valid MXCANCEL values.

If the sending nucleus does not receive notification that the canceled nucleus has terminated within the MXCANCEL timeout period, it does one of the following things:

- It returns response code 124 (ADARSP124), subcode 28 (no reply) to the user on whose behalf the intracluster command was issued.

- It terminates itself abnormally if the communication occurred on behalf of an internal process that must not fail.

When a cluster nucleus fails, the other nuclei in the cluster can recover from the failure only once they know for sure that the first nucleus has disconnected from the shared structures and is no longer writing to the database. When an unresponsive cluster nucleus has been canceled, it is important that it terminate quickly, so that it does not hold up the recovery process performed by the other nuclei in the cluster.

**Note:**

Do not specify an MPMDUMP JCL statement for a cluster nucleus. An MPMDUMP is written before the failing nucleus disconnects from the shared structures and may significantly delay the start of the online recovery process performed by the other nuclei in the cluster. Instead, on z/OS systems, consider specifying a //SVCDUMP DD DUMMY statement, which usually produces a very quick dump prior to disconnecting. On other systems, and as a backup method on z/OS, specify a regular system dump statement (e.g., SYSUDUMP on z/OS) for a dump that is produced after disconnecting.

The value for MXCANCEL can be changed dynamically (for the time the Adabas nucleus is running) using the MXCANCEL operator command.

## MXCANCELWARN Parameter: Timeout Threshold for Canceled Peer Nucleus Warning

Parameter	Specify . . .	Minimum	Maximum	Default
MXCANCELWARN	the timeout threshold, in seconds, for an intracluster cancellation request to occur before an early warning about the failure of the cancellation request is issued.	0 (no warning) or 4	MXCANCEL - 1	MXCANCEL / 4

The optional MXCANCELWARN parameter specifies the number of seconds after an intracluster cancellation request is made when the cluster nucleus requesting the cancellation should issue warnings about the inability of the canceled peer nucleus to terminate. If no value is specified for the MXCANCELWARN parameter, the value used is calculated as the value of the ADARUN MXCANCEL parameter divided by four. When dividing by four to determine the default value, the quotient is rounded up to the next integer value.

The value of MXCANCELWARN must be less than the value of MXCANCEL. If MXCANCELWARN is set to a value greater than or equal to the value specified for MXCANCEL, the value is rejected. If MXCANCELWARN is set to a value of zero (0), no warnings are issued about canceled cluster nuclei that are slow to terminate.

If the target nucleus does not terminate within the time period set by MXCANCELWARN, the nucleus that issued the cancellation request writes message ADAX9G to the operator console, identifying the target nucleus by its nucleus ID, job name, and system name. This message indicates that the nucleus writing the message may be in danger of terminating itself if the canceled peer nucleus does not terminate within the timeout period defined by the MXCANCEL parameter. You can use the ADAX9G message to trigger an alert or take other appropriate action before the MXCANCEL timeout period expires.

If the peer nucleus targeted for cancellation terminates after the ADAX9G message is written, the nucleus that issued the cancellation request writes message ADAX9K to the operator console, indicating that the cancellation occurred and the requesting nucleus is no longer in danger of terminating itself as a result of this particular incident.

The value for MXCANCELWARN can be changed dynamically (for the time the Adabas nucleus is running) using the MXCANCELWARN operator command.

**Note:**

If the value of the MXCANCEL parameter is dynamically changed using the MXCANCEL operator command while the nucleus is running, but its new value is less than or equal to the current MXCANCELWARN setting, the value of MXCANCELWARN is automatically set to zero. You can then use the MXCANCELWARN operator command to dynamically set the MXCANCELWARN parameter to a new value that is less than the new MXCANCEL value.

## MXMSG Parameter: Timeout Threshold for Internucleus Command Processing

Parameter	Specify . . .	Minimum	Maximum	Default
MXMSG	the timeout threshold, in seconds, in which the processing of an intracenter command must occur.	15	32767	300

The optional MXMSG parameter specifies the maximum number of seconds in which one cluster nucleus waits for the response of another nucleus in the cluster to an internal, intracenter command. If a value for MXMSG is not specified, a value of 300 seconds (5 minutes) is used.

If the receiving nucleus does not respond to the intracenter command within the time period set by the MXMSG parameter, the requesting cluster nucleus issues a cancellation request for the unresponsive nucleus. The cancellation is requested to preserve the ability of all of the other active nuclei in the cluster to process user commands.

Adabas nuclei working on the same cluster database must collaborate to keep the database physically and logically consistent while processing user commands. To do this, they intermittently submit internal, intracenter commands to one another. If a nucleus in the cluster does not respond to these intracenter requests in a timely manner, the other nuclei in the cluster may quickly or gradually become unable to continue processing user commands.

Problems that may prevent a cluster nucleus from responding to intracenter commands fall into two broad categories:

- Problems have occurred in the cluster nucleus, such as a CPU loop or a resource deadlock. In such cases, the nucleus experiencing the problem must be terminated to resolve the problem and the sooner this happens, the sooner the other cluster nuclei can recover from the failure and get back to normal processing.
- Problems have occurred outside the cluster nucleus, such as CPU starvation or a problem in the host system. In such cases, the cause of the delayed response might possibly be removed by system measures unrelated to the cluster database, and letting the other cluster nuclei wait some time for a resolution might prevent an unnecessary abnormal termination of a cluster member.

The value of MXMSG should strike a balance between these two categories, so that faulty cluster nuclei that cannot possibly stay active are canceled reasonably quickly while cluster nuclei that are victims of temporary system problems are not unnecessarily terminated. For assistance in setting the value of the MXMSG parameter, review the statistics provided on the messaging performance reports produced at termination or by the DXMSG operator command.

The value for MXMSG can be changed dynamically (for the time the Adabas nucleus is running) using the MXMSG operator command.

## MXMSGWARN Parameter: Timeout Threshold for Internucleus Command Processing Warning

Parameter	Specify . . .	Minimum	Maximum	Default
MXMSGWARN	the timeout threshold, in seconds, for a response to an intracluster command to occur before an early warning about the failure of the intracluster request is issued	0 (no warning) or 4	MXMSG - 1	MXMSG / 4

The optional MXMSGWARN parameter specifies the number of seconds after an intracluster request is made when the cluster nucleus issuing the intracluster request should issue warnings about the failure of the target cluster nucleus to respond. If no value is specified for the MXMSGWARN parameter, the value used is calculated as the value of MXMSG divided by four. When dividing by four to determine the default value, the quotient is rounded up to the next integer value.

The value of MXMSGWARN must be less than the value of MXMSG. If MXMSGWARN is set to a value greater than or equal to the value specified for MXMSG, the value is rejected. If MXMSGWARN is set to zero, no warnings are issued about cluster nuclei that are slow to respond.

If the target nucleus does not respond within the time period set by MXMSGWARN, the nucleus that issued the request writes message ADAX9C to the operator console, identifying the target nucleus by its nucleus ID, job name, and system name. This message indicates that the target nucleus may be in danger of being canceled by the nucleus that issued the request if it does not respond within the timeout period defined by the MXMSG parameter. You can use the ADAX9C message to trigger an alert or take other appropriate action before the MXMSG timeout period expires.

If the target nucleus responds to the intracluster command after the ADAX9C message is written, the nucleus that issued the intracluster-command writes message ADAX9D to the operator console, indicating that a response was given and the target nucleus is no longer in danger of being canceled as a result of this particular incident.

The value for MXMSGWARN can be changed dynamically (for the time the Adabas nucleus is running) using the MXMSGWARN operator command.

### Note:

If the value of the MXMSG parameter is dynamically changed using the MXMSG operator command while the nucleus is running, but its new value is less than or equal to the current MXMSGWARN setting, the value of MXMSGWARN is automatically set to zero. You can then use the MXMSGWARN operator command to dynamically set the MXMSGWARN parameter to a new value that is less than the new MXMSG value.

## MXWTOR Parameter: Self-Termination Operator Query Interval

Parameter	Specify . . .	Minimum	Maximum	Default
MXWTOR	the timeout interval, in seconds, in which an operator response to a self-termination operator query must occur.	0 (no warning) or 15	64800	0

The optional MXWTOR parameter specifies the maximum number of seconds that a cluster nucleus should wait for an operator response to a self-terminating operator query. If no value is specified for the MXWTOR parameter, a value of zero is used. Values from 1 through 14 are not valid MXWTOR values.

A cluster nucleus issues a self-terminating operator query to the console via message ADAX9J if a canceled peer nucleus remains active after the time period defined by the MXCANCEL parameter. The MXWTOR parameter specifies how long the nucleus should wait for a response to this query.

One way that the operator may reply to the ADAX9J message is to make sure that the canceled cluster nucleus mentioned in the message actually terminates; once this happens, the ADAX9J message is retracted automatically, an online recovery process is performed, and the nucleus that issued the ADAX9J message remains active.

If the operator does not respond to the ADAX9J message within the time period set by the MXWTOR parameter and the canceled peer nucleus has not terminated in the meantime, the nucleus that requested the operator response and printed message ADAX9J terminates itself with message ADAX99.

If MXWTOR is set to zero (the default), the operator query does not occur (no ADAX9J messages are written) and no operator intervention is expected. Instead, the nucleus terminates itself immediately with the ADAX99 message and user abend code 79.

Some installations may want to specify MXWTOR value long enough to allow the operator a good opportunity to bring down the canceled peer nucleus. Other installations may prefer not to get such operator queries at all.

The value for MXWTOR can be changed dynamically (for the time the Adabas nucleus is running) using the MXWTOR operator command.

## NUCID Parameter: Adabas Cluster Nucleus ID

Parameter	Specify . . .	Possible Values	Default
NUCID	the ID for a nucleus in an Adabas cluster.	0-65000	0

NUCID identifies an Adabas cluster nucleus. The cluster may span operating system images under Adabas Cluster Services, or it may be confined to a single operating system image under Adabas Parallel Services. A value greater than 0 (the default) identifies this nucleus as an Adabas cluster nucleus.



Value	Meaning
0	The default setting. This is not an Adabas cluster nucleus.
1-65000	This is an Adabas cluster nucleus. This value must be unique among all target IDs declared to the Adabas router (ADASVC): Adabas databases (DBID), cluster nuclei (NUCID), Natural buffer pools and other MPM servers, Entire Net-Work, and internal targets used by Adabas Cluster Services and Adabas Parallel Services (restricted range 65,001-65,479). Nucleus IDs for the same cluster need not be specified contiguously.

### Guidelines

- Each nucleus being run must have its own Work data set.
- All nuclei in a cluster must be run against the same database.
- Software AG strongly recommends that you run utilities or user applications against a nucleus in the local z/OS image. Although a remote open nucleus may be used, performance degradation due to network overhead is likely.

### Example

The following example starts an Adabas cluster session with the specified cluster nuclei:

```
ADARUN PROG=ADANUC,NUCID=1, ...
```

## PROGRAM Parameter: Program to Run

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

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

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

## Examples

The following example specifies that an Adabas nucleus is running.

```
ADARUN PROGRAM=ADANUC
```

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

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

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

```
ADARUN PROGRAM=NETWRK
```

## SMFDETAIL Parameter: Selecting Detail SMF Record Sections

Parameter	Specify . . .	Valid Values	Default
SMFDETAIL	the type of detail sections in SMF Interval and Termination records (subtypes 2 and 3) that should be included in Adabas SMF records. All specifications (regardless of how many) for this parameter should be enclosed in parentheses.	NONE  ALL  comma-delimited list of detail section names	NONE

Adabas SMF records can contain a variable set of detail sections in Interval and Termination records (subtypes 2 and 3). This parameter allows you to select the detail section types in Interval and Termination records that should be included in the Adabas SMF records. The value or values you specify for this parameter must be enclosed in parentheses. If you specify NONE or ALL, they should be specified alone. However, you can specify one or more of the other detail section names (CMD, CSHB, CSHF, CSHG, CSHP, FILE, IODD, LOCK, MSGB, MSGC, MSGH, PARM, STG, THRD, or USER) in one SMFDETAIL parameter, separating each value with a comma.

### Note:

The CSHB, CSHF, CSHG, LOCK, MSGB, MSGC, and MSGH detail section names apply only to cluster environments (Adabas Cluster Services or Adabas Parallel Services must be installed). In addition, the CSHP detail section name only applies in cluster environments with Adabas Parallel Services installed.

The following table describes the meaning of the possible detail section names that can be used in the SMFDETAIL parameter:

Detail Section Name	Description
ALL	Generate all possible detail sections in the Adabas SMF records. If this value is specified, no others can be specified on the SMFDETAIL parameter.
CMD	Generate Adabas command activity detail sections in the Adabas SMF records.
CSHB <sup>1</sup>	Generate global cache activity by block detail sections in the Adabas SMF records.
CSHF <sup>1</sup>	Generate global cache activity by Adabas file number detail sections in the Adabas SMF records.
CSHG <sup>1</sup>	Generate global cache activity detail sections in the Adabas SMF records.

Detail Section Name	Description
CSHP <sup>2</sup>	Generate Adabas Parallel Services cache activity detail sections in the Adabas SMF records.
FILE	Generate Adabas file activity detail sections in the Adabas SMF records.
IODD	Generate I/O activity by DD statement detail sections in the Adabas SMF records.
LOCK <sup>1</sup>	Generate global lock activity detail sections in the Adabas SMF records.
MSGB <sup>1</sup>	Generate internucleus messaging control block activity detail sections in the Adabas SMF records.
MSGC <sup>1</sup>	Generate internucleus messaging count detail sections in the Adabas SMF records.
MSGH <sup>1</sup>	Generate internucleus messaging service time histogram detail sections in the Adabas SMF records.
NONE	Generate no detail sections in the Adabas SMF records. If this value is specified, no others can be specified on the SMFDETAIL parameter.
PARM	Generate ADARUN parameter value detail sections in the Adabas SMF records.
STG	Generate Adabas storage pool detail sections in the Adabas SMF records.
THRD	Generate thread activity detail sections in the Adabas SMF records.
USER	Generate user-defined details sections in the Adabas SMF records. If USER is specified, a value for the UEXSMF parameter should also be specified to identify the user exit to be invoked to generate the user-defined detail section.
<sup>1</sup> Available only in cluster environments (Adabas Cluster Services or Adabas Parallel Services must be installed).	
<sup>2</sup> Available only in cluster environments with Adabas Parallel Services installed.	

This parameter can be altered by operator commands while the nucleus is running.

### Example

In the following example, detail sections for Adabas command, file, and thread activity are generated and written in the Adabas SMF records.

```
ADARUN PROG=ADANUC, SMFDETAIL=(CMD, FILE, THRD)
```

In the following example, all detail sections are generated and written in the Adabas SMF records.

```
ADARUN PROG=ADANUC, SMFDETAIL=(ALL)
```

## V64BIT Parameter: 64-Bit Virtual Storage Option

### Note:

This parameter is applicable only to z/OS systems.

Parameter	Specify . . .	Possible Values	Default
V64bit	whether or not the Adabas nucleus should use virtual storage above the two gigabyte bar (64-bit virtual storage). In order for the nucleus to use 64-bit virtual storage, the operating system must also support 64-bit virtual storage.	YES   NO	NO

This parameter indicates whether or not the Adabas nucleus should use 64-bit virtual storage. The default is NO.

Currently, the following storage areas can be allocated above the two gigabyte bar in an Adabas nucleus:

- The flush I/O pool (see ADARUN parameter LFIOP)
- The PLOG protection I/O buffers (see ADARUN parameter NPLOGBUFFERS)
- The Work part 1 protection I/O buffers (see ADARUN parameter NWORK1BUFFERS)
- The cast-out directory buffer, in cluster environments (automatically derived from the attributes of the global cache space for Adabas Cluster or Adabas Parallel Services).

The use of 64-bit virtual storage will help alleviate virtual storage constraints below the two gigabyte bar in installations that use large amounts of virtual storage in Adabas nucleus sessions (for example, nucleus sessions with ADARUN parameter settings of LBP=1200M and LFIOP=200M).

Value	Meaning
YES	The Adabas nucleus should use 64-bit virtual storage (storage above the two gigabyte bar).
NO	The default setting. The Adabas nucleus will not use 64-bit virtual storage.

Virtual storage above the two gigabyte bar (64-bit virtual storage) is shown in dumps to SVCDUMP or SYSDUMP but not in dumps to MPMDUMP, SYSABEND, or SYSUDUMP. Therefore, we recommend that you use the SVCDUMP in the JCL of an Adabas nucleus running with ADARUN V64BIT=YES to ensure that 64-bit virtual storage is shown in any SVC dumps that are produced. If, for some reason, you do not want to produce SVC dumps, we recommend that you use SYSDUMP to ensure 64-bit virtual storage is shown in a dump of the Adabas nucleus address space.

## Example

The following example indicates that the Adabas nucleus should make use of virtual storage above the 2 gigabyte bar and should use large pages.

```
ADARUN PROG=ADANUC,LARGEPAGE=YES,V64BIT=YES
```