Operator Commands

Adabas operator commands are entered during an Adabas session or during utility operation to

- terminate an Adabas or user session;

- display nucleus or utility information;

- log commands into CLOG;

- change Adabas operating parameters or conditions.

In this chapter, the commands are listed alphabetically. One command, DSTAT, is listed twice: once as a command for displaying nucleus status, and separately as a command to display current Adabas utility operating status.

- For Adabas Caching Facility operator commands, see the Adabas Caching Facility documentation.

- For Adabas Parallel Services operator commands, see the Adabas Parallel Services documentation.

- For operator commands in a sysplex cluster environment, see the Adabas Cluster Services documentation.

This chapter covers the following topics:

- Entering Operator Commands

- Operator Command Groupings

- Nucleus Commands

- Adabas Utility Operator Command DSTAT

- SMGT Operator Command

- SMGT Display Output Samples

Entering Operator Commands

This section describes how to enter operator commands in different environments. It is divided into the following topics:

- Entering Commands on BS2000 Systems

- Entering Commands on z/OS Systems

- Entering Commands on z/VSE Systems

Entering Commands on BS2000 Systems

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

```
/INTR   TSN, command
```

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

```
/INTR   command
```

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

Operator commands are processed by a STXIT routine.

Entering Commands on z/OS Systems

To enter operator commands in z/OS environments, use the z/OS MODIFY (F) command as shown below:
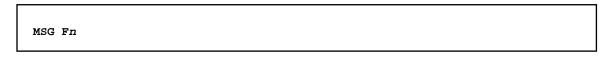
```
F  jobname,command
```

where *jobname* is the name specified by the EXEC job control statement (usually ADARUN).

Entering Commands on z/VSE Systems

To enter operator commands in z/VSE environments, use the following steps:

1. Enter an MSG command for the z/VSE partition in which Adabas is executing.

   Example:

   ```
   MSG Fn
   ```

   When ready for communication, Adabas will respond with Adabas message ADAI29. In cases where the commands ADAEND, CANCEL, and HALT cause Adabas to end the nucleus session, no outstanding reply is presented so that an orderly shutdown can occur without the need for operator intervention.

2. Enter the desired Adabas operator command or commands.

   More than one command can be entered in a session. You can also enter a command followed directly by a slash (/) to end any further operator communications until the next MSG command by z/VSE.

3. When all desired commands have been entered, close the operator communication session by entering a null command (EOB).

4. To enter any more commands after the session is closed, repeat this entire procedure.

Operator Command Groupings

The Adabas operator commands are grouped as follows:

- *Nucleus Commands*

- *Adabas Utility Operator Command DSTAT*

- *SMGT Operator Command*

Nucleus Commands

The following operator commands can be entered from the console to monitor and control Adabas nucleus operation.

- ADAEND Command

- ALOCKF Command

- AOSLOG Command

- ASYTVS Command

- CANCEL Command

- CLOGMRG Command

- CT Command

- DAUQ Command

- DCQ Command

- DDIB Command

- DDSF Command

- DELUF Command

- DELUI Command

- DFILES Command

- DFILUSE Command

- DHQ Command

- DHQA Command

- DLOCKF Command

- DNC Command

- DNFV Command

- DNH Command

- DNU Command

- DONLSTAT Command

- DPARM Command

- DPPT Command

- DRES Command

- DSTAT Command

- DTH Command

- DUQ Command

- DUQA Command

- DUQE Command

- DUUQE Command

- FEOFCL Command

- FEOFPL Command

- FMXIO Command

- HALT Command

- LICREFRESH Command

- LOCKF Command

- LOCKU Command

- LOCKX Command

- LOGGING Command
- LOGCB Command
- LOGFB Command
- LOGIB Command
- LOGIO Command
- LOGRB Command
- LOGSB Command
- LOGUX Command
- LOGVB Command
- LOGVOLIO Command
- LOGWARN Command
- NOLOGGING Command
- NOLOGCB Command
- NOLOGFB Command
- NOLOGIB Command
- NOLOGIO Command
- NOLOGRB Command
- NOLOGSB Command
- NOLOGUX Command
- NOLOGVB Command
- NOLOGVOLIO Command
- NWCONNECT Command
- ONLRESUME Command
- ONLSTOP Command
- ONLSUSPEND Command>
- RALOCKF Command
- RALOCKFA Command

- RDUMPST Command

- READONLY Command

- REVIEW Command

- SMFDETAIL Command

- SMFDETAILADD Command

- SMFDETAILDEL Command

- SMFINTERVAL Command

- SMFRECNO Command

- SMFSUBSYS Command

- STOPF Command

- STOPI Command

- STOPU Command

- SYNCC Command

- TCPIP Command

- TNAA Command

- TNAE Command

- TNAX Command

- TT Command

- UNLOCKF Command

- UNLOCKU Command

- UNLOCKX Command

- UTIONLY Command

Note:
The DUMP command, which stopped nucleus operation and issued a dump, is no longer supported. To stop operation, use an operating system-dependent command such as a z/OS Cancel.

ADAEND Command

ADAEND [,GLOBAL]

Use the ADAEND command to terminate Adabas session normally. No new users are accepted after this command has been issued. ET logic updating is continued until the end of the current logical transaction for each user. After all activity has been completed as described above, the Adabas session is terminated. In nucleus cluster environments, the GLOBAL option can be used to terminate the Adabas session in all active cluster nuclei.

ALOCKF Command

ALOCKF=n

Use the ALOCKF command to lock a file in advance to ensure that an EXU, EXF, or UTI user will be able to obtain exclusive control of the specified file. The advance-lock prevents new transactions from using the file. Once all current users have stopped using the file, the exclusive-control user will get the lock. Until then, Adabas keeps the exclusive-control user waiting.

To remove the advance lock without running the utility, see the RALOCKF command.

This command is not available in single user mode or for a read-only nucleus. It is available in cluster and non-cluster environments.

The following key points should be noted about advance-locks on files:

1. An advance-lock can be set while a file is being used.

2. A command requesting exclusive control (UTI, EXF, or EXU) over an advance-locked file will wait in the command queue until all other users stop using the file before it starts processing the file.

3. Advance-locks are automatically removed when a user gets exclusive control over the file. However, if a file is locked (via the LOCKF, LOCKU, or LOCKX commands), the locks are not removed when a user gets exclusive control over the file. (Locks must be explicitly removed, whereas advance-locks are automatically removed.)

4. Adabas will reject an advance-lock on a file that is already locked (via the LOCKF, LOCKU, or LOCKX commands) but will accept a lock request on an advance-locked file.

5. To ensure you have uninterrupted exclusive control over a file in a situation where you have multiple steps to run that require uninterrupted exclusive control while all steps have been processed, use a combination of advance-locking the file (ALOCKF), stopping all users of the file (STOPF), and locking the file (LOCKU).

6. In the case of expanded files, an ALOCKF command is applied to the anchor file (representing the entire expanded file chain).

7. In a cluster environment, advance-locks are effective in all nuclei of the cluster.

AOSLOG Command

AOSLOG={YES | NO}

Use the AOSLOG command to activate and deactivate logging of certain Adabas calls that modify the nucleus to DD/PRINT. These calls are issued either by ADADBS OPERCOM or Adabas Online System. Read and display calls are not logged.

ASYTVS Command

```
ASYTVS={YES | NO}
```

Use the ASYTVS command to activate or deactivate asynchronous flushing of buffers based on volume serial number.

CANCEL Command

```
CANCEL [,GLOBAL]
```

Use the CANCEL command to cancel the Adabas session immediately. All command processing is immediately suspended. A pending autorestart will be in effect which in turn will cause the autorestart routine to be executed during the initialization of the next Adabas session. In nucleus cluster environments, the GLOBAL option can be used to cancel the Adabas session in all active cluster nuclei.

CLOGMRG Command

```
CLOGMRG={YES | NO}
```

Use the CLOGMRG command to dynamically modify the setting of the ADARUN CLOGMRG parameter.

The CLOGMRG command is only valid in cluster environments. It is global by definition and affects all nuclei in the cluster.

CT Command

```
CT=timeout-limit
```

Use the CT command to dynamically override the ADARUN CT parameter value; that is, the maximum number of seconds that can elapse from the time an Adabas command has been completed until the results are returned to the user through interregion communication (which depends on the particular operating system being used). The minimum setting is 1; the maximum is 16777215.

In nucleus cluster environments, the CT command is global by definition and affects all nuclei in the cluster.

DAUQ Command

```
DAUQ
```

Use the DAUQ command to display the user queue elements of those users who have executed at least one Adabas command within the last 15 minutes.

DCQ Command

```
DCQ
```

Use the DCQ command to display all posted command queue elements (CQEs). The DCQ command displays each CQE's user ID, job name, and buffer length.

DDIB Command

```
DDIB
```

Use the DDIB command to display the data integrity block (DIB). This block contains entries indicating which Adabas utilities are active and the resources being used by each utility.

DDSF Command

```
DDSF
```

Use the DDSF command to display Adabas Delta Save Facility status. The DDSF command is only available if the Adabas nucleus is run with the parameter ADARUN DSF=YES.

DELUF Command

```
DELUF=file-number
```

Use the DELUF command to delete all users who are using the specified file. Any open transactions of the deleted users are backed out.

This command does not delete EXF or UTI users.

The DELUF command corresponds to the ADADBS OPERCOM STOPF=*file-number*,PURGE function.

Caution:
If Adabas is running with ADARUN OPENRQ=NO (specifying that users are not required to issue an OP as the first command of the session), run the DELUF command only if you are certain that the users to be deleted are no longer active. If a user with an open transaction is deleted, but then returns (by sending a command), no indication is given about the transaction backout. If the user continues the transaction, logical inconsistencies in the database could occur.

DELUI Command

DELUI= *timeout-limit*

Use the DELUI command to delete all users who have not executed a command during the specified time interval (in seconds). Any open transactions of the deleted users are backed out.

This command does not delete EXF or UTI users.

The DELUI command corresponds to the ADADBS OPERCOM STOPI=*time*,PURGE function.

Caution:
If Adabas is running with ADARUN OPENRQ=NO (specifying that users are not required to issue an OP as the first command of the session), run the DELUI command only if you are certain that the users to be deleted are no longer active. If a user with an open transaction is deleted, but then returns (by sending a command), no indication is given about the transaction backout. If the user continues the transaction, logical inconsistencies in the database could occur.

DFILES Command

DFILES= { *n* | *n1* ,..., *n5* }

Use the DFILES command to display the number of users currently accessing, updating, or controlling either a specific file (*n*) or a series of individual files, specified in a list (*n1*,...,*n5*). A maximum of five files can be specified in the list. Users are displayed by job name and Adabas-assigned user ID, and listed by file.

DFILUSE Command

DFILUSE= *nnn*

Use the DFILUSE command to display the number of total commands processed so far for the specified file during the current session. The count is displayed in the nucleus message ADAN33.

DHQ Command

```
DHQ
```

Use the DHQ command to display up to five hold queue elements.

DHQA Command

```
DHQA
```

Use the DHQA command to display up to 1000 hold queue elements.

DLOCKF Command

```
DLOCKF
```

Use the DLOCKF command to display the locked files.

DNC Command

```
DNC
```

Use the DNC command to display the number of posted command queue elements waiting to be selected.

DNFV Command

```
DNFV
```

Use the DNFV command to display information about current file use.

This command provides information about the files in use at a particular point in time. It also indicates which other nucleus has exclusive file control if, for example, a user program receives a response 148 (ADARSP148), subcode 15.

Here is some sample output:

```
ADAI29 OPER CMD: DNFV

FNR=00008  A=Y U=Y ID=      CA=00000   CU=00001
```

where;

FNR=*nnnnn*	is the file number
A={Y | N}	(yes or no) indicates whether the file is used for access (read and/or search)
U={Y | N}	(yes or no) indicates whether the file is used for update. Use for update includes use for access.
ID=*nucid*	is the ID of the nucleus that owns the file lock, if the file is locked.
CA=*nnnnn*	is the number of users on this nucleus who are currently accessing this file.
CU=*nnnnn*	is the number of users on this nucleus who are currently updating this file.

DNH Command

Use the DNH command to display the number of ISNs currently in the hold queue.

DNU Command

Use the DNU command to display the number of current users.

DONLSTAT Command

Use the DONLSTAT command to display the status of each active reorder, invert online, or Event Replicator for Adabas initial-state process together with the process ID.

DPARM Command

Use the DPARM command to display the Adabas session parameters currently in effect. Here is an example:

```
ADAI29 Oper cmd: DPARM
ADAN16 00226 2009-07-06 17:09:03 READONLY=NO,UTIONLY=NO
ADAN16 00226 2009-07-06 17:09:03 ASYTVS=YES,AOSLOG=NO
ADAN16 00226 2009-07-06 17:09:03 NC=200,NH=500,NT=20,NU=200
ADAN16 00226 2009-07-06 17:09:03 NPLOGBUFFERS=1
```

```
ADAN16 00226 2009-07-06 17:09:03 NWORK1BUFFERS=1
ADAN16 00226 2009-07-06 17:09:03 LBP=376064,LFP=12000,LWP=500000
ADAN16 00226 2009-07-06 17:09:03 LI=10000,LP=200,LQ=10000,LS=49920
ADAN16 00226 2009-07-06 17:09:03 LFIOP=100000
ADAN16 00226 2009-07-06 17:09:03 FMXIO=1,LU=164000
ADAN16 00226 2009-07-06 17:09:03 TT=900,TNAA=900,TNAE=900,CT=60
ADAN16 00226 2009-07-06 17:09:03 TNAX=900,MXTT=3600,MXTNA=3600
ADAN16 00226 2009-07-06 17:09:03 TLSCMD=300,MXTSX=3600
ADAN16 00226 2009-07-06 17:09:03 V64BIT=NO,LARGEPAGE=NO
ADAN16 00226 2009-07-06 17:09:03 NOLOG
ADAN16 00226 2009-07-06 17:09:03 LOGVOLIO=NO
ADAN41 00226 2009-07-06 17:09:03 Function completed
```

Additional Adabas add-on products and nucleus components may display more parameters than are shown in this sample. For example, the DPARM command includes settings for the ADARUN parameters related to Adabas Cluster Services and Adabas Parallel Services alert and timeout statistics.

```
17:28:14 ADAI29 Oper cmd: DPARM
17:28:14 ADAN16 00226 2007-06-01 17:28:13 READONLY=NO,UTIONLY=NO
17:28:14 ADAN16 00226 2007-06-01 17:28:13 ASYTVS=YES,AOSLOG=NO
17:28:14 ADAN16 00226 2007-06-01 17:28:13 NC=200,NH=500,NT=20,NU=200
17:28:14 ADAN16 00226 2007-06-01 17:28:13 LBP=375808,LFP=12000,LWP=500000
17:28:14 ADAN16 00226 2007-06-01 17:28:13 LI=10000,LP=200,LQ=10000,LS=49920
17:28:14 ADAN16 00226 2007-06-01 17:28:13 LFIOP=100000,FMXIO=1,LU=65535
17:28:14 ADAN16 00226 2007-06-01 17:28:13 TT=900,TNAA=900,TNAE=900,CT=60
17:28:14 ADAN16 00226 2007-06-01 17:28:13 TNAX=900,MXTT=3600,MXTNA=3600
17:28:14 ADAN16 00226 2007-06-01 17:28:13 TLSCMD=300,MXTSX=3600
17:28:14 ADAN16 00226 2007-06-01 17:28:13 NOLOG
17:28:14 ADAN16 00226 2007-06-01 17:28:13 NUCID=2261,MXMSG=300,MXMSGWARN=60
17:28:14 ADAN16 00226 2007-06-01 17:28:13 MXCANCEL=600,MXCANCELWARN=120
17:28:14 ADAN16 00226 2007-06-01 17:28:13 MXWTOR=0
17:28:14 ADAN16 00226 2007-06-01 17:28:13 CLUSTER=SYSPLEX,CLUGROUPNAME=PTGCJP
17:28:14 ADAN16 00226 2007-06-01 17:28:13 CLULOCKNAME=ADA_LOCK7
17:28:14 ADAN16 00226 2007-06-01 17:28:13 CLUCACHENAME=ADA_CACHE7
17:28:14 ADAN16 00226 2007-06-01 17:28:13 DIRRATIO=4,ELEMRATIO=1,LRDP=100000
17:28:14 ADAN16 00226 2007-06-01 17:28:13 CLUCACHEEXTRA=2000
17:28:14 ADAN41 00226 2007-06-01 17:28:13 Function completed
```

DPPT Command

Use the DPPT command to display the parallel participant table (PPT) block for a nucleus.

This command produces internal information for use by Software AG technical support.

Sample Output

```
ADAI29 Oper cmd: DPPT
ADAN24 00199 2009-01-27 22:31:35 Display PPT RABNs 00000083 to 000000A2
ADAN24 00199 2009-01-27 22:31:35
ADAN24 00199 2009-01-27 22:31:35          PPT RABN: 00000083
ADAN24 00199 2009-01-27 22:31:35 Number of entries: 05
ADAN24 00199 2009-01-27 22:31:35 Nucleus indicator: E2
ADAN24 00199 2009-01-27 22:31:35             NUCID: 0000
ADAN24 00199 2009-01-27 22:31:35    Session number: 0005
ADAN24 00199 2009-01-27 22:31:35   Last PLOG block: 00000000
```

```
ADAN24 00199 2009-01-27 22:31:35    PLOG block ind: 00
ADAN24 00199 2009-01-27 22:31:35  PPT Entry length: 0023
ADAN24 00199 2009-01-27 22:31:35          Entry ID: E6
ADAN24 00199 2009-01-27 22:31:35 Dataset=/SAGUID/PLX2/V7/WORKR1/
ADAN24 00199 2009-01-27 22:31:35  PPT Entry length: 0023
ADAN24 00199 2009-01-27 22:31:35          Entry ID: 61
ADAN24 00199 2009-01-27 22:31:35 Dataset=/ SAGUID /PLX2/V7/PLOGR1/
ADAN24 00199 2009-01-27 22:31:35  PPT Entry length: 0023
ADAN24 00199 2009-01-27 22:31:35          Entry ID: 62
ADAN24 00199 2009-01-27 22:31:35 Dataset=/ SAGUID /PLX2/V7/PLOGR2/
ADAN24 00199 2009-01-27 22:31:35  PPT Entry length: 0023
ADAN24 00199 2009-01-27 22:31:35          Entry ID: 41
ADAN24 00199 2009-01-27 22:31:35 Dataset=/ SAGUID /PLX2/V7/CLOGR1/
ADAN24 00199 2009-01-27 22:31:35  PPT Entry length: 0023
ADAN24 00199 2009-01-27 22:31:35          Entry ID: 42
ADAN24 00199 2009-01-27 22:31:35 Dataset=/ SAGUID /PLX2/V7/CLOGR2/
ADAN41 00199 2009-01-27 22:31:35 Function completed
```

DRES Command

DRES

Use the DRES command to display the allocated pool space and the highest use level (high water mark) reached so far during the current session by record count and by percent for the following resources:

- Attached buffers (AB)

- Command queue (CQ)

- Format pool (FP)

- Hold queue (HQ)

- Pool for the table of ISNs (TBI)

- Pool for the table of sequential commands (TBQ or TBLES)

- User queue (UQ)

- Unique descriptor pool (DUQPOOL)

- Security pool

- Replication pool

- User queue file list pool

- Work pool (WP)

- Pool for global transaction IDs (XIDs; nonzero only with Adabas Transaction Manager)

- Redo pool (nonzero only with Adabas Cluster Services)

- Work part 1 area (WKP1)

  Note:
  The maximum pool value of Work part 1 is derived from the LP parameter. It corresponds to the maximum number of blocks a transaction can spend on Work Part 1 before Adabas decides to back it out.

- Work part 2 area (WKP2)

- Work part 3 area (WKP3)

The actual values are displayed in nucleus message ADAN28.

DSTAT Command

```
DSTAT
```

Use the DSTAT command to display the current Adabas nucleus status.

DTH Command

```
DTH
```

Use the DTH command to display thread status.

DUQ Command

```
DUQ
```

Use the DUQ command to display up to five active and inactive user queue elements.

DUQA Command

```
DUQA
```

Use the DUQA command to display up to 100 user queue elements.

DUQE Command

```
DUQE= X'userid'
```

Use the DUQE command to display the user queue element for the specified Adabas-assigned user ID. The user ID must be entered in hexadecimal format as follows:

```
DUQE=X'A3C1F2'
```

Do not enter a job name in place of the user ID.

DUUQE Command

DUUQE

Use the DUUQE command to display utility user queue elements.

FEOFCL Command

FEOFCL [,GLOBAL]

Use the FEOFCL command to close the current dual or multiple command log and switch to the another command log. This command is valid only if dual or multiple command logging is in effect.

In nucleus cluster environments, the GLOBAL option can be used to close and switch dual or multiple command logs in all active cluster nuclei.

FEOFPL Command

FEOFPL [,GLOBAL]

Use the FEOFPL command to close the current dual or multiple data protection log and switch to the another protection log. This command is valid only if dual or multiple data protection logging is in effect.

In nucleus cluster environments, the GLOBAL option can be used to close and switch dual or multiple protection logs in all active cluster nuclei.

FMXIO Command

FMXIO=*nn*

Use the FMXIO command to dynamically modify the setting of the ADARUN FMXIO parameter.

HALT Command

```
HALT [,GLOBAL]
```

Use the HALT command to perform a BT (back out transaction) for each active ET logic user's session, then terminate the Adabas session. No dumps are produced by HALT.

In nucleus cluster environments, the GLOBAL option can be used to perform a BT for all active ET logic user sessions and terminate the Adabas session in all active cluster nuclei.

LICREFRESH Command

```
LICREFRESH
```

Use the LICREFRESH command to:

- reload the license module or reread the license file from the library identified by the DDLIC JCL statement in the startup job for the nucleus

- display and check the license identified by the DDLIC JCL statement in the startup job for the nucleus.

In cluster environments, the LICREFRESH command must be run on each nucleus in the cluster.

LOCKF Command

```
LOCKF=n
```

Use the LOCKF command to lock the specified file. No use of the specified file is possible at any security level.

LOCKU Command

```
LOCKU=n
```

Use the LOCKU command to lock the specified file for all non-utility users. Adabas utilities can use the specified file normally.

LOCKX Command

```
LOCKX=n
```

Use the LOCKX command to lock the specified file for all users except EXU or EXF users. EXU and EXF users can use the file normally. The lock is released automatically when an EXU user issues an OP command.

LOGGING Command

```
LOGGING
```

Use the LOGGING command to start command logging.

LOGCB Command

```
LOGCB
```

Use the LOGCB command to start logging of the Adabas control block for each command logged.

LOGFB Command

```
LOGFB
```

Use the LOGFB command to start logging of the Adabas format buffer for each command logged.

LOGIB Command

```
LOGIB
```

Use the LOGIB command to start logging of the Adabas ISN buffer for each command logged.

LOGIO Command

```
LOGIO
```

Use the LOGIO command to start logging of Adabas I/O activity for each command logged.

LOGRB Command

```
LOGRB
```

Use the LOGRB command to start logging of the Adabas record buffer for each command logged.

LOGSB Command

```
LOGSB
```

Use the LOGSB command to start logging of the Adabas search buffer for each command logged.

LOGUX Command

```
LOGUX
```

Use the LOGUX command to start logging of user exit B data for inclusion in the CLOG record. This command is only valid when CLOGLAYOUT=5.

LOGVB Command

```
LOGVB
```

Use the LOGVB command to start logging of the Adabas value buffer for each command logged.

LOGVOLIO Command

```
LOGVOLIO
```

Use the LOGVOLIO command to initiate writing of the extended I/O list to the command log for CLOGLAYOUT=5 and CLOGLAYOUT=8.

LOGWARN Command

```
LOGWARN = { seconds | 0 }
```

Use the LOGWARN command to specify how often the PLOG and CLOG status is checked and resulting alert messages are produced. Valid values range from zero (0) through 2147483647 seconds. The default is 0, indicating that no PLOG or CLOG status checking occurs and no corresponding alert messages are produced. If a non-zero value is specified for LOGWARN, a valid user exit 2 or user exit 12 must also be specified.

NOLOGGING Command

```
NOLOGGING
```

Use the NOLOGGING command to stop or prevent command logging.

NOLOGCB Command

```
NOLOGCB
```

Use the NOLOGCB command to stop or prevent logging of the Adabas control block.

NOLOGFB Command

```
NOLOGFB
```

Use the NOLOGFB command to stop or prevent logging of the Adabas format buffer.

NOLOGIB Command

```
NOLOGIB
```

Use the NOLOGIB command to stop or prevent logging of the Adabas ISN buffer.

NOLOGIO Command

```
NOLOGIO
```

Use the NOLOGIO command to stop or prevent logging of Adabas I/O activity.

NOLOGRB Command

```
NOLOGRB
```

Use the NOLOGRB command to stop or prevent logging of the Adabas record buffer.

NOLOGSB Command

```
NOLOGSB
```

Use the NOLOGSB command to stop or prevent logging of the Adabas search buffer.

NOLOGUX Command

```
NOLOGUX
```

Use the NOLOGUX command to stop logging of user exit B data for inclusion in the CLOG record. This command is only valid when CLOGLAYOUT=5.

NOLOGVB Command

```
NOLOGVB
```

Use the NOLOGVB command to stop or prevent logging of the Adabas value buffer.

NOLOGVOLIO Command

```
NOLOGVOLIO
```

Use the NOLOGVOLIO command to terminate (stop) writing the extended I/O list to the command log for CLOGLAYOUT=5 and CLOGLAYOUT=8.

NWCONNECT Command

```
NWCONNECT
```

Use the NWCONNECT command to retry establishing the Entire Net-Work target associated with the nucleus's IDT entry. For classig Adabas nuclei, this is the DBID; for Adabas Cluster Services or Adabas Parallel Services, this is the nucleus ID (NUCID).

During nucleus initialization, certain Entire Net-Work errors may have prevented the target from being established, resulting in message ADAM76. Other errors may prevent Entire Net-Work from defining the target when it is started after the nucleus is initialized. The Entire Net-Work operator command DISPLAY TARGETS may be used to see whether the DBID target is known to Entire Net-Work.

Note:
Adabas Cluster Services and Adabas Parallel Services maintain the Entire Net-Work DBID target using a different protocol. The nucleus will attempt to reestablish the DBID target automatically at timed intervals.

ONLRESUME Command

ONLRESUME=X'*identifier*'

Use the ONLRESUME command to resume a previously suspended online reorder, invert, or Event Replicator for Adabas initial-state process.

ONLSTOP Command

ONLSTOP=X'*identifier*'

Use the ONLSTOP command to stop an online reorder, invert, or Event Replicator for Adabas initial-state process cleanly. The process continues up to its next interrupt point in order to produce a consistent state, and then terminates after performing all necessary cleanup.

ONLSUSPEND Command>

ONLSUSPEND=X'*identifier*'

Use the ONLSUSPEND command to suspend an online reorder, invert, or Event Replicator for Adabas initial-state process. The process continues up to its next interrupt point in order to produce a consistent state, performs a command throwback, and enters a state where it cannot be selected for processing. This command is useful if the online process is consuming too much of the nucleus resources.

RALOCKF Command

RALOCKF=*n*

Use the RALOCKF command to release the advance-lock on the specified file (see ALOCKF command) without running the utility.

This command is available in cluster and non-cluster environments.

RALOCKFA Command

```
RALOCKFA
```

Use the RALOCKFA command to release the advance-lock on all files for which it has been set (see ALOCKF command) without running the utility.

This command is available in cluster and non-cluster environments.

RDUMPST Command

```
RDUMPST
```

Use the RDUMPST command to terminate online dump status. This command is normally used if online execution of the ADASAV utility has terminated abnormally.

READONLY Command

```
READONLY={YES | NO}
```

Use the READONLY command to switch READONLY status on or off. A value of "YES" switches it on; a value of NO switches it off.

REVIEW Command

```
REVIEW={ NO | LOCAL | hub-id }
```

Use the REVIEW command to:

- deactivate Adabas Review,
- change from hub mode to local mode, or
- to specify or change the Adabas Review hub with which a nucleus communicates.

SMFDETAIL Command

```
SMFDETAIL = { ALL | NONE | section-name-list }
```

Adabas SMF records can contain a variable set of detail sections in Interval and Termination records (subtypes 2 and 3). This command allows you to override the setting of the SMFDETAIL ADARUN parameter defined for this Adabas sesssion. Using this command you can select the detail section types in Interval and Termination records that should be included in the Adabas SMF records. Unlike the SMFDETAIL ADARUN parameter, the value or values you specify for the SMFDETAIL command do *not* need to 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.

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

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 [1]	Generate global cache activity by block detail sections in the Adabas SMF records.
CSHF [1]	Generate global cache activity by Adabas file number detail sections in the Adabas SMF records.
CSHG [1]	Generate global cache activity detail sections in the Adabas SMF records.
CSHP [2]	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 [1]	Generate global lock activity detail sections in the Adabas SMF records.
MSGB [1]	Generate internucleus messaging control block activity detail sections in the Adabas SMF records.
MSGC [1]	Generate internucleus messaging count detail sections in the Adabas SMF records.
MSGH [1]	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.

Detail Section Name	Description
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.

[1] *Available only in cluster environments (Adabas Cluster Services or Adabas Parallel Services must be installed).*

[2] *Available only in cluster environments with Adabas Parallel Services installed.*

SMFDETAILADD Command

SMFDETAILADD = *section-name-list*

Adabas SMF records can contain a variable set of detail sections in Interval and Termination records (subtypes 2 and 3). This command allows you to add specific detail sections to your Adabas SMF records for the running Adabas session. The sections you specify are added to those already specified for the Adabas sesssion. If more than one section is listed on an SMFDETAILADD command, separate the section names with commas.

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

Detail Section Name	Description
CMD	Generate Adabas command 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 recods.
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 section sin the Adabas SMF records. If USER is specified, a value for the UEXSMF ADARUN parameter can also be specified to identify the user exit that should be used to generate the user-defined detail section.

SMFDETAILDEL Command

SMFDETAILDEL = *section-name-list*

Adabas SMF records can contain a variable set of detail sections in Interval and Termination records (subtypes 2 and 3). This command allows you to remove specific detail sections from your Adabas SMF records for the running Adabas session. The sections you specify are removed from those already specified for the Adabas sesssion. If more than one section is listed on an SMFDETAILDEL command, separate the section names with commas.

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

Detail Section Name	Generates
CMD	Adabas command activity detail sections in the Adabas SMF records.
FILE	Adabas file activity detail sections in the Adabas SMF records.
IODD	I/O activity by DD statement detail sections in the Adabas SMF recods.
PARM	ADARUN parameter value detail sections in the Adabas SMF records.
STG	Adabas storage pool detail sections in the Adabas SMF records.
THRD	Thread activity detail sections in the Adabas SMF records.
USER	Uuser-defined details section sin the Adabas SMF records.

SMFINTERVAL Command

SMFINTERVAL = { GLOBAL | NONE | SUBSYS | *minutes* **}**

Adabas SMF records can be generated at specific intervals. The SMFINTERVAL command enables and disables interval recording and specifies the interval or the source from which the interval can be derived.

The interval can be one of several values from z/OS specifications or an explicit interval in minutes. Interval records contain a product detail section and whatever detail sections are selected in the SMFDETAIL parameter, except for the ADARUN parameter detail section which are not included in SMF interval records.

The following table describes the meaning of the specifications that can be used in the SMFINTERVAL command:

Valid Value	Description
GLOBAL	Generate all interval SMF records at the rate established by the INTVL and SYNCVAL SMF parameters in PARMLIB member SMFPRM*xx*.
NONE	Generate no interval SMF records.
SUBSYS	Generate interval SMF records at the rate specified in PARMLIB member SMFPRM*xx* for the subsystem identified in the ADARUN SMFSUBSYS parameter.
minutes	Generate interval SMF records at the specified interval, in minutes. Valid minute values can range from 1 through 9999..

SMFRECNO Command

```
SMFRECNO = record-number
```

Use this command to override the SMFRECNO setting currently specified for this Adabas session (either through the SMFRECNO ADARUN parameter or by another SMFRECNO command). This command sets the Adabas SMF record number used for user-defined SMF records. IBM designates the range of SMF numbers from 128 through 255 for user-defined records, so valid values range from 128 through 255.

SMFSUBSYS Command

```
SMFSUBSYS = { CURRENT | SMF-subsystem-name }
```

This command overrides any other SMFSUBSYS setting active in the Adabas session (either via the ADARUN SMFSUBSYS parameter or another SMFSUBSYS command). It allows you to associate the Adabas SMF records with an IBM or user-defined subsystem. The z/OS systems staff can provide different sets of SMF system parameters for IBM-defined subsystems. Up to eight user-specified subsystems can be defined, each with their own set of system parameters. This parameter allows you to optionally identify one of these subsystem names to be used with Adabas SMF recording, if the one you want is different from the current one.

The possible values for this command are:

- CURRENT: This is the default value and associates the Adabas SMF record with the IBM-defined subsystem under which the Adabas nucleus is active. The subsystem may be TSO for a nucleus running in a TSO session, STC for a starting task or the name of the job entry subsystem under which a batch job is running, JES2 or JES3.

- An explicit subsystem name can be specified. It may be either an IBM-defined name or a user-defined name from PARMLIB member SMFPRM*xx*. The subsystem name is one to four characters long. The first character must be alphabetic or national (#, @, or $) and the remaining characters can be either alphanumeric or national characters. Contact your z/OS administrator for more information about the subsystem names available in your z/OS environment.

STOPF Command

```
STOPF=file-number
```

Use the STOPF command to stop all users who are using the specified file. Any open transactions of the stopped users will be backed out. A stopped user who returns (by sending a command) will receive response code 9 (ADARSP009).

This command does not stop EXF or UTI users.

STOPI Command

STOPI= *time*

Use the STOPI command to stop all users who have not executed a command during the specified time interval (in seconds). Any open transactions of the stopped users will be backed out. A stopped user who returns (by sending a command) will receive response code 9 (ADARSP009).

This command does not stop EXF or UTI users.

STOPU Command

$$\text{STOPU=} \begin{Bmatrix} X'userid' \\ job\text{-}name \end{Bmatrix}$$

Use the STOPU command to stop and delete the user with the Adabas-assigned user ID (in the form shown in the display commands), or stop and delete all users with the specified job name (*job-name*). Any open transactions of the stopped users will be backed out.

Caution:
If Adabas is running with ADARUN OPENRQ=NO (specifying that users are not required to issue an OP as the first command of the session), run the STOPU command only if you are certain that the users to be deleted are no longer active. If a user with an open transaction is deleted, but then returns (by sending a command), no indication is given about the transaction backout. If the user continues the transaction, logical inconsistencies in the database could occur.

Note:
The STOPU=X'*userid*' command is not allowed for online reorder or invert processes. Use the ONLSTOP=X'*identifier*' command instead.

The user ID must be specified in hexadecimal format; for example:

```
STOPU=X'1CF2'
```

SYNCC Command

SYNCC

Use the SYNCC command to force synchronization for all ET users. The nucleus waits for all ET users to reach ET status.

TCPIP Command

```
TCPIP={ OPEN=url | CLOSE=url | CLOSE }
```

Use the TCPIP command to open or close a direct TCP/IP link to the Adabas nucleus or to close all TCP/IP links when no URL is specified.

This command is only possible when the ADARUN parameter `TCPIP` is set to "YES" and all conditions for that setting have been met. This command can be used to close the URL set in the ADARUN TCPURL parameter, or to open or close additional TCP/IP links.

You must identify the universal resource locator (URL) for the TCP/IP link you want to open or close. The URL is a 20-byte address that conforms to the RFC specification for URLs:

api-name://stackid:port-number

where:

api-name	is a 1-3 character value identifying the application programming interface (API) to use. Both APIs for the IBM TCP/IP stack (HPS, OES) and the API for the Interlink stack (ILK) are currently supported.
stackid	is a 1-8 character value identifying the stack to use:

- for the HPS API, this is the name of the TCP/IP started task.

- for the OES API, no value is needed.

- for the ILK API, this is the subsystem identifier.

port-number	is a 1-5 character number in decimal notation.

Examples

```
TCPIP=OPEN=ILK://ILZ5:1234
```

```
TCPIP=CLOSE=ILK://ILZ5:1234
```

To close all open URLs:

```
TCPIP=CLOSE
```

TNAA Command

```
TNAA=n
```

Use the TNAA command to set the non-activity time limit for access-only users. This value must be greater than zero and replaces the value set by the TNAA ADARUN parameter.

In nucleus cluster environments, the TNAA command is global by definition and affects all nuclei in the cluster.

TNAE Command

```
TNAE=n
```

Use the TNAE command to set the non-activity time limit for ET logic users. This value must be greater than zero and replaces the value set by the TNAE ADARUN parameter.

In nucleus cluster environments, the TNAE command is global by definition and affects all nuclei in the cluster.

TNAX Command

```
TNAX=n
```

Use the TNAX command to set the non-activity time limit for exclusive control users. This value must be greater than zero and replaces the value set by the TNAX ADARUN parameter.

In nucleus cluster environments, the TNAX command is global by definition and affects all nuclei in the cluster.

TT Command

```
TT=n
```

Use the TT command to set the transaction time limit for ET logic users. This value must be greater than zero and replaces the value set by the TT ADARUN parameter.

In nucleus cluster environments, the TT command is global by definition and affects all nuclei in the cluster.

UNLOCKF Command

```
UNLOCKF=n
```

Use the UNLOCKF command to unlock the specified file. File usage is restored to its pre-locked status.

UNLOCKU Command

```
UNLOCKU=n
```

Use the UNLOCKU command to unlock the specified file that was previously locked for all non-utility users. File usage is restored to its pre-locked status.

UNLOCKX Command

```
UNLOCKX=n
```

Use the UNLOCKX command to unlock the specified file that was previously locked for non-exclusive control users. File usage is restored to its pre-locked status.

UTIONLY Command

```
UTIONLY={YES | NO}
```

Use the UTIONLY command to switch the ADARUN UTIONLY status parameter on or off. The default is NO.

Adabas Utility Operator Command DSTAT

```
DSTAT
```

Use the DSTAT command to display the current ADALOD, ADAORD, ADARES, ADASAV, ADAULD, or ADAVAL operating status. The following are examples of the output that results when DSTAT is entered during specific Adabas utility operations:

Note:
ADAORD, ADARES, ADAULD, and ADAVAL do not accept operator commands if they run in MODE=SINGLE.

Example 1: ADALOD

```
ADAU00 dbid OPERATOR TYPE-IN : DSTAT ADAU22 dbid LOADING DATA STORAGE. REC-NO=3599
ADAU00 dbid OPERATOR TYPE-IN : DSTAT ADAU21 dbid SORTING/LOADING DESCRIPTOR AA
```

Example 2: ADAORD

```
ADAU08 dbid OPERATOR TYPE-IN : DSTAT ADAU26 dbid UNLOADING INDEX. FILE=1
ADAU08 dbid OPERATOR TYPE-IN : DSTAT ADAU25 dbid UNLOADING DATASTORAGE. FILE=2
ADAU08 dbid OPERATOR TYPE-IN : DSTAT ADAU29 dbid LOADING DATASTORAGE. FILE=2
```

Example 3: ADARES

```
ADAU08 dbid OPERATOR TYPE-IN : DSTAT REGENERATE INPUT VOLUME = BMC002, PLOG-NUM = 12
FROMBLK =         1, FROMTIME = 1996-04-10 11:27:56
TOBLK =           1, TOTIME   = 1996-04-10 11:27:56
```

Example 4: ADASAV

```
ADAU08 dbid OPERATOR TYPE-IN : DSTAT ADAU92 dbid       STILL  INITIALIZING
ADAU08 dbid OPERATOR TYPE-IN : DSTAT ADAU10 dbid 435 BLOCKS OUT OF 465 SAVED
ADAU08 dbid OPERATOR TYPE-IN : DSTAT ADAU11 dbid 342 BLOCKS OUT OF 451 RESTORED
```

Example 5: ADAULD

```
ADAU08 dbid OPERATOR TYPE-IN : DSTAT ADAU67 dbid UNLOADING FILE=17, RECNO=2875
```

SMGT Operator Command

The Adabas error handling and message buffering facility uses a single operator command, SMGT, followed by a comma and one or more operands:

Valid *operands* are: ABNORMAL, ACTPIN, ADDPIN, DEACTPIN, DELPIN, DISPLAY, DUMP, MSGBUF, OFF, ON, SNAP, XACTIVATE, XCRITICAL, XDEACTIVATE, XLOAD, and XNOTCRITICAL. Some of these operands are mutually exclusive. All operands are described in *Operands for SMGT*.

Operands may require that you enter one or more of the following variables:

Variable Type	Description
exit-code	The code that identifies an exit routine to an SMGT operator command is one of the following: ● UEX*n*, where *n* is a single-digit user exit number ● UX*nn*, where *nn* is a double-digit user exit number ● HX*nn*, where *nn* is a hyperexit number (single-digit numbers are preceded by a zero; e.g., HX02) ● CX00 (the Adabas Caching Facility user exit) ● SX00 (the PIN routine user exit)
start, *end*	The range of addresses for the SNAP operand where *start* is the hexadecimal address where the SNAP dump begins and *end* is the hexadecimal address where it ends.
module-name	The name of the module. For the XLOAD operand, it is the name of the exit routine module to be loaded; for ADDPIN and DELPIN operands, it is the name of the PIN module to be added or deleted, respectively.
pin-number	When a PIN is added, it is assigned a number that can be found using the DISPLAY=PINS operand. This pin number is used for the ACTPIN and DEACTPIN operands to identify the PIN routine to be activated or deactivated, respectively.

Acceptable abbreviations for operands are provided in mixed case notation (capital letters required, lowercase letters optional). Default values for operands are underlined.

Operands for SMGT

SMGT,{ ABNormalterm | MSGbuf } = { ON | OFF }

The ABNORMALTERM (ABN) operand determines whether the error handling and message buffering facility will handle abnormal termination errors.

The MSGBUF (MSG) operand temporarily deactivates (OFF) or reactivates (ON) message buffering.

The ABN and MSG operands are mutually exclusive; if one is specified on an individual SMGT command, the other cannot be.

SMGT,{ ACTpin | DEACTpin } = *pin-number*

The ACTPIN (ACT) operand is used to activate an individual PIN routine or to reactivate it after it has been temporarily deactivated.

The DEACTPIN (DEACT) operand is used to deactivate an individual PIN routine.

The ACT and DEACT operands are mutually exclusive; if one is specified on an individual SMGT command, the other cannot be.

SMGT,{ ADDpin | DELpin } = *module-name*

The ADDPIN (ADD) operand adds PIN routines to the error handling facility. The PIN routine module indicated is loaded and the PINs that are found in it are added to the facility. When a PIN is added, it is assigned a number which can be found using the DISPLAY=PINS operand.

The DELPIN (DEL) operand deletes a PIN module and all the PINs it contains from the facility.

The ADD and DEL operands are mutually exclusive; if one is specified on an individual SMGT command, the other cannot be.

SMGT,Display = { ALL | EXITS | MSGBUF | PINS | SUMMARY | LAST }

The DISPLAY (D) operand writes status and history information about the error handling and message buffering facility to the job log and to DDPRINT:

ALL	(the default) displays all reports
EXITS	displays current user exit, hyperexit, and other exit status
MSGBUF	displays message buffering status
PINS	displays PIN routine status
SUMMARY	displays only the summary information from ALL
LAST	displays details of the most recent recovery action

Sample output for the various DISPLAY values is provided in *SMGT Display Output Samples*.

SMGT,DUMP = { ON | OFF }

The DUMP operand determines whether a full system dump is taken for the Adabas nucleus in the event of an error. The default (OFF) means that only a snap dump is taken.

Note:
To use this command successfully under z/OS, the data set ADASNAP must be provided in the Adabas startup JCL. For more information, read *Adabas Session Execution*.

```
SMGT,{ ON | OFF }
```

The OFF operand deactivates the error handling and message buffering facility temporarily; the ON operand reactivates it.

When SMGT=OFF:

- The only valid SMGT operands that can be specified after SMGT=OFF is specified are ON (in a new SMGT command) and DISPLAY. All other SMGT commands are rejected until SMGT=ON is issued.

- All other functions of the error handling facility are disabled. PIN modules cannot be invoked. Any user exits marked NOTCRITICAL are treated as CRITICAL; that is, the nucleus terminates abnormally if an error occurs in the exit.

The ON and OFF operands are mutually exclusive; if one is specified on an individual SMGT command, the other cannot be.

```
SMGT,SNAP [= ( start , end ) ]
```

The SNAP operand displays a formatted dump of the nucleus without error diagnostics. If SNAP is specified without any additional parameters, the whole nucleus is displayed; if hexadecimal addresses are specified for start and end, the SNAP dump is displayed only for that range of addresses.

The SNAP operand can degrade system performance as long as it is active.

Note:
To use this command successfully under z/OS, the data set ADASNAP must be provided in the Adabas startup JCL. For more information, read *Adabas Session Execution*.

```
SMGT,{ XActivate | XDeactivate } = exit-code
```

The XACTIVATE (XA) operand activates a loaded exit module; the XDEACTIVATE (XD) operand deactivates an active exit module.

The XA and XD operands are mutually exclusive; if one is specified on an individual SMGT command, the other cannot be.

```
SMGT,{ XNotcritical | XCritical } = exit-code
```

The XNOTCRITICAL (XN) operand changes the status of an exit from the default critical to noncritical for the functioning of the Adabas nucleus. Any abnormal termination or program check in a noncritical exit results in the exit being automatically deactivated; however, the Adabas nucleus continues to run. The disabled user exit is not recalled until it is reset. Once the exit error has been corrected, the exit can be

reactivated using the XACTIVATE operand.

The XCRITICAL (XC) operand changes the status of an exit from noncritical back to the default critical for the functioning of the Adabas nucleus. Any abnormal termination or program check in a critical exit causes the Adabas nucleus to terminate.

The XN and XC operands are mutually exclusive; if one is specified on an individual SMGT command, the other cannot be.

$$\text{SMGT,XLoad} = \begin{Bmatrix} \textit{exit-code} \\ (\textit{exit-code}, \textit{module-name}) \end{Bmatrix}$$

The XLOAD (XL) operand loads a new version of an exit module.

The module-name parameter is required only for new exits. Because the default is the previous module name, the parameter is optional for exits that have previously been used. The parameter is also optional for any exit defined with a number of zero; for example, the ADACSH exit which is always named ADACSHUX.

SMGT Display Output Samples

This section provides sample output for the DISPLAY operand of the SMGT command.

SMGT,Display = { ALL | EXITS | MSGBUF | PINS | SUMMARY | LAST }

The DISPLAY (D) operand writes status and history information about the Adabas error handling and message buffering facility to the job log and to DDPRINT:

ALL	(the default) displays all reports
EXITS	displays current user exit, hyperexit, and other exit status
MSGBUF	displays message buffering status
PINS	displays PIN routine status
SUMMARY	displays only the summary information from ALL
LAST	displays details of the most recent recovery action

This section provides sample output for the following options of the DISPLAY operand:

- DISPLAY=ALL

- DISPLAY=EXITS

- DISPLAY=MSGBUF

- DISPLAY=PINS

- DISPLAY=SUMMARY

- DISPLAY=LAST

DISPLAY=ALL

The ALL parameter displays all of the information shown for the other parameters in the following paragraphs.

DISPLAY=EXITS

```
ADANA1 00127 SMGT DISPLAY ACTIVE
ADANA2 00127 SMGT ACTIVE
ADANAX 00127 EXIT: UX04 MODNAME: EXIT1     STATUS: ACTIVE
ADANAX 00127 EXIT: HX22 MODNAME: NULLEXIT STATUS: ACTIVE
ADANO2 00127 SMGT COMMAND PROCESSED
```

DISPLAY=MSGBUF

```
ADANA1 00127 SMGT DISPLAY ACTIVE
ADANA2 00127 SMGT ACTIVE
ADANAE 00127 MESSAGE BUFFERING IS ACTIVE
ADANO2 00127 SMGT COMMAND PROCESSED
```

DISPLAY=PINS

```
ADANA1 00127 SMGT DISPLAY ACTIVE
ADANA2 00127 SMGT ACTIVE
ADANAG 00127 PIN 0001 USES: 0000 CONDITION: 000C4000
             THIS PIN VALID FOR ALL LOCATIONS
ADANAG 00127 PIN 0002 USES: 0000 CONDITION: 000C1000
             LOCATION: 00081C6C 0008259B (EBL2    )
ADANAA 00127 002 CONDITION PIN ROUTINES RECOVERED 000 ERRORS
ADANAB 00127 000 LOCATION PIN ROUTINES RECOVERED 000 ERRORS
ADANAG 00127 PIN 0003 USES: 0000 CONDITION: RSP: 017
             THIS PIN VALID FOR ALL LOCATIONS
ADANAC 00127 001 RESPONSE PIN ROUTINES RECOVERED 000 ERRORS
ADANAD 00127 003 TOTAL PIN ROUTINES RECOVERED 000 ERRORS
ADANA8 00127 000 EXECUTIONS OF ABNORMAL TERMINATION HANDLER
ADANA8 00127 000 EXECUTIONS OF PROGRAM CHECK HANDLER
ADANA8 00127 000 EXECUTIONS OF RESPONSE CODE HANDLER
ADANO2 00127 SMGT COMMAND PROCESSED
```

DISPLAY=SUMMARY

```
ADANA1 00127 SMGT DISPLAY ACTIVE
ADANA2 00127 SMGT ACTIVE
ADANA3 00127 ABNORMAL TERMINATION HANDLER ACTIVE
ADANA4 00127 PROGRAM CHECK HANDLER ACTIVE
ADANAA 00127 002 CONDITION PIN ROUTINES RECOVERED 000 ERRORS
ADANAB 00127 000 LOCATION PIN ROUTINES RECOVERED 000 ERRORS
ADANAC 00127 001 RESPONSE PIN ROUTINES RECOVERED 000 ERRORS
ADANAD 00127 003 TOTAL PIN ROUTINES RECOVERED 000 ERRORS
ADANA8 00127 000 EXECUTIONS OF ABNORMAL TERMINATION HANDLER
ADANA8 00127 000 EXECUTIONS OF PROGRAM CHECK HANDLER
ADANA8 00127 000 EXECUTIONS OF RESPONSE CODE HANDLER
ADANO2 00127 SMGT COMMAND PROCESSED
```

DISPLAY=LAST

No error:

```
ADANA1 00127 SMGT DISPLAY ACTIVE
ADANA2 00127 SMGT ACTIVE
ADANA5 00127 NO ERROR CONDITIONS HANDLED
ADANO2 00127 SMGT COMMAND PROCESSED
```

Error Encountered:

```
ADANA1 00127 SMGT DISPLAY ACTIVE
ADANA2 00127 SMGT ACTIVE
ADANA6 00127 LAST ERROR OCCURRED AT: 1998-07-07 14:36:18
ADANA7 00127 CONDITION: RSP: 017 LOCATION: *N/A*
ADANO2 00127 SMGT COMMAND PROCESSED
```