

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 Commands
 - SMGT Display Output Samples
-

Entering Operator Commands

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.

OS/390 or z/OS Systems

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

```
F jobname,command
```

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

VM/ESA or z/VM Systems

In VM/ESA environments, use the CP SEND command to enter operator commands from either the console of the virtual machine or from a secondary console of a disconnected virtual machine.

Example:

```
CP SEND USER04 ADAEND
```

- issued at the DBA virtual machine console sends the ADAEND command to the USER04 virtual machine, providing that the CP directory for the USER04 virtual machine contains a CONSOLE statement enabling the DBA console as a secondary console.

VSE/ESA Systems

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

1. Enter an MSG command for the VSE/ESA 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 VSE/ESA.

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 Commands

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.

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 an OS Cancel.

```
ADAEND [,GLOBAL]
```

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=n
```

Lock a file in advance to ensure that an EXU, EXF, or UTI user will have 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 has the lock. Until then, the exclusive-control user must wait.

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.

AOSLOG={YES | NO}

Activate/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/display calls are not logged.

ASYTVS={YES | NO}

Activate/deactivate asynchronous flushing of buffers based on volume serial number.

CANCEL [,GLOBAL]

Cancel 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={YES | NO}

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.

CLUFREEUSER=[,TNA= *max-time*][,UID | UIDPREFIX] = *userid*][,FORCE][,GLOBAL]

Note:

The CLUFREEUSER command is only valid in cluster environments. It can be issued against the local nucleus only or, with the GLOBAL option, against all active and inactive nuclei in the cluster.

Delete leftover cluster user table elements (UTEs or PLXUSERS) in common storage that are no longer associated with user queue elements (UQEs) in a nucleus where

TNA is a decimal number specifying the timeout value in seconds. PLXUSERS that are not used during the time specified may be deleted if other conditions are fulfilled. If TNA= is not specified, PLXUSERS may be deleted without regard to their recent use.

UID	is a character string or hexadecimal byte string format as follows:
cccccccc	where the argument is 1-8 letters, digits, or embedded '-' signs without surrounding apostrophes.
'cccccccc'	where the argument is 1-8 characters with surrounding apostrophes.
X'xxxxxxxxxxxxxxxx'	where the argument is an even number of 2-16 hexadecimal digits enclosed by X' '.

A character string must be enclosed in apostrophes if it contains characters other than letter, digits, or embedded '-' signs. If a specified character string is less than eight characters long, it is implicitly padded with blanks. If a specified hexadecimal string is shorter than 16 hexadecimal digits, it is implicitly padded with binary zeros. If the last eight bytes of a user's 28-byte communication ID match a specific user ID or user ID prefix (UIDPRFX parameter), that user's PLXUSER may be deleted if other conditions are fulfilled. If UID is not specified, PLXUSERS may be deleted regardless of their user IDs. "UID" may be abbreviated to "U".

UID and UIDPRFX are mutually exclusive; one or the other, but not both, can be specified.

UIDPRFX UIDPRFX accepts the same operands as UID. It operates in the same manner as UID, except that the operand is considered a prefix and there is no implicit padding. If the operand matches the initial bytes of the last eight bytes of the 28-byte communication ID, that PLXUSER may be deleted if other conditions are fulfilled. "UIDPRFX" may be abbreviated to "UP".

UID and UIDPRFX are mutually exclusive; one or the other, but not both, can be specified.

FORCE Leftover PLXUSERS are to be deleted even if the users are due a response code 9, subcode 20. If FORCE is not specified, such PLXUSERS are not deleted. Before using the FORCE parameter, ensure that the users owning the PLXUSERS to be deleted will not expect any of their transactions to remain open.

GLOBAL Leftover PLXUSERS throughout the Adabas cluster are to be deleted if they are no longer associated with UQEs and are eligible according to the other specified parameters. Additionally and subject to the other rules, leftover PLXUSERS are deleted if their assigned nuclei have terminated since their last use. If GLOBAL is not specified, only PLXUSERS assigned to the local nucleus and used since the nucleus start are eligible for deletion.

CT=*timeout-limit*

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

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

DCQ

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

DDIB

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

DDSF

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

DELUF=*file-number*

Delete all user queue elements for the specified file. The DELUF command corresponds to the ADADBS OPERCOM STOPF=*file-number*,PURGE function.

DELUI=*timeout-limit*

Delete all user queue elements that remained inactive for the specified amount of time (specified in seconds). The DELUI command corresponds to the ADADBS OPERCOM STOPI=time,PURGE function.

```
DFILES={ n | n1-nx | n1,..., n5 }
```

Display the number of users currently accessing, updating, or controlling either file "n", a range of files "n1 - nx" , or files "n1,...,n5". A maximum of five files can be specified. Users are displayed by job name and Adabas-assigned user ID, and listed by file.

```
DFILUSE=nnn
```

Display the count of total commands processed so far for the specified file during the current session. The count is displayed in the nucleus message ADAN33.

```
DHQ
```

Display up to five hold queue elements.

```
DHQA
```

Display up to 1000 hold queue elements.

```
DLOCKF
```

Display locked files.

```
DNC
```

Display the number of posted command queue elements.

```
DNFV
```

Use the DNFV command to display nucleus file variables; that is, 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, subcode 15.

Sample Output

```
ADAI29 OPER CMD: DNFV
FNR=00008 A=Y U=Y ID=          CA=00000 CU=00001
```

where;

FNR= <i>nnnnn</i>	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= <i>nucid</i>	is the ID of the nucleus that owns the file lock, if the file is locked.
CA= <i>nnnnn</i>	is the number of users on this nucleus who are currently accessing this file.
CU= <i>nnnnn</i>	is the number of users on this nucleus who are currently updating this file.

DNH

Display the number of ISNs currently in the hold queue.

DNU

Display the number of current users.

DONLSTAT

Display the status of each active reorder, invert online, or Event Replicator for Adabas initial-state process together with the process ID.

DPARM

Use the DPARM command to display the Adabas session parameters currently in effect.

Sample Output

```
ADAI29 Oper cmd: DPARM
ADAN16 00199 2009-01-27 22:48:43 READONLY=NO,UTIONLY=NO
ADAN16 00199 2009-01-27 22:48:43 ASYTVS=YES,AOSLOG=NO
ADAN16 00199 2009-01-27 22:48:43 NC=100,NH=500,NT=8,NU=100
ADAN16 00199 2009-01-27 22:48:43 LBP=5026304,LFP=20000,LWP=320000
ADAN16 00199 2009-01-27 22:48:43 LI=10000,LP=500,LQ=10000,LS=79872
ADAN16 00199 2009-01-27 22:48:43 LFIOP=80000,FMXIO=1,LU=65535
ADAN16 00199 2009-01-27 22:48:43 TT=6000,TNAA=6000,TNAE=6000,CT=60
ADAN16 00199 2009-01-27 22:48:43 TNAX=6000,MXTT=3600,MXTNA=3600
ADAN16 00199 2009-01-27 22:48:43 TLSCMD=300,MXTSX=3600
ADAN16 00199 2009-01-27 22:48:43 NOLOG
ADAN41 00199 2009-01-27 22:48:43 Function completed
```



DPPT

Use the DPPT command to display the parallel participant table (PPT); that is, to display the nucleus's own PPT block.

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

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) - current allocation not supported
- 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)
- cluster block update "redo" pool (nonzero only with Adabas Cluster Services)

The actual values are displayed in nucleus message ADAN28 described in the Adabas Messages and Codes documentation.

DSTAT

Display the current Adabas nucleus status.

DTH

Display thread status.

DUQ

Display up to five active and inactive user queue elements.

DUQA

Display up to 100 user queue elements.

DUQE=X*userid*

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

Display utility user queue elements.

FEOFCL [,GLOBAL]

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 [,GLOBAL]

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=nn

Dynamically modify the setting of the ADARUN FMXIO parameter.

HALT [,GLOBAL]

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.

LOCKF=*n*

Lock the specified file. No use of the specified file is possible at any security level.

LOCKU=*n*

Lock the specified file for all non-utility users. Adabas utilities can use the specified file normally.

LOCKX=*n*

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

Begin command logging.

LOGCB

Begin logging of the Adabas control block for each command logged.

LOGFB

Begin logging of the Adabas format buffer for each command logged.

LOGIB

Begin logging of Adabas ISN buffer for each command logged.

LOGIO

Begin logging of Adabas I/O activity for each command logged.

LOGRB

Begin logging of the Adabas record buffer for each command logged.

LOGSB

Begin logging of the Adabas search buffer for each command logged.

LOGUX

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

LOGVB

Begin logging of the Adabas value buffer for each command logged.

NOLOGGING

Stop or prevent command logging.

NOLOGCB

Stop or prevent logging of the Adabas control block.

NOLOGFB

Stop or prevent logging of the Adabas format buffer.

```
NOLOGIB
```

Stop or prevent logging of the Adabas ISN buffer.

```
NOLOGIO
```

Stop or prevent logging of Adabas I/O activity.

```
NOLOGRB
```

Stop or prevent logging of the Adabas record buffer.

```
NOLOGSB
```

Stop or prevent logging of the Adabas search buffer.

```
NOLOGUX
```

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

```
NOLOGVB
```

Stop or prevent logging of the Adabas value buffer.

```
ONLRESUME=X'identifier'
```

Resume a previously-suspended online reorder, invert, or Event Replicator for Adabas initial-state process.

```
ONLSTOP=X'identifier'
```

Stops 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=X*identifier*
```

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=n
```

Remove the advance lock on the specified file (see ALOCKF command) without running the utility.

```
RALOCKFA
```

Remove the advance lock on all files for which it has been set (see ALOCKF command) without running the utility.

```
RDUMPST
```

Terminate online dump status. This command is normally used if online execution of the ADASAV utility has terminated abnormally.

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

Switches READONLY status on or off.

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

Deactivate Adabas Review; change from hub mode to local mode; specify or change the Adabas Review hub with which a nucleus communicates.

```
STOPF=file-number
```

Stop users who are using the specified file.

This command does not stop EXF or UTI users.

```
STOPI= time
```

Stop users who have not been executing during a given period of time (in seconds).

This command does not stop EXF or UTI users.

```
STOPU= { X'userid'  
        job-name }
```

Note:

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

Stop the user with the Adabas-assigned user ID (in the form shown in the display commands), or stop all users found with the job "*job-name*".

STOPU/STOPF/STOPI clear inactive or timed-out users and deletes the user queue element (UQE) for each identified user. If the program/user is an ET logic user, is not in ET status, and has not been stopped before one of these commands (STOPU, STOPF, or STOPI) is issued, Adabas backs out all updates made by the transaction to this point and releases all held records. If the transaction continues, only those changes following the STOPU, STOPF, or STOPI command are completed.

Note:

Before stopping a user, ensure that the user is in fact no longer active. If the stopped user returns (by sending a command), no indication is given about the transaction backout.

The user ID must be specified in hexadecimal format:

```
STOPU=X'A3C1F2'
```

```
SYNCC
```

Force synchronization for all ET users. The nucleus waits for all ET users to reach ET status.

```
TCP/IP={ OPEN=url | CLOSE=url | CLOSE }
```

Open or close a direct TCP/IP link to the Adabas nucleus or 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/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= n
```

Set 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= n
```

Set 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.



TNAE=*n*

Set 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.



TNAX=*n*

Set 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=*n*

Unlock the specified file. File usage is restored to its pre-locked status.



UNLOCKU=*n*

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



UNLOCKX=*n*

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



UTIONLY={YES | NO}

Switch UTIONLY status on or off. The default is NO.

Adabas Utility Operator Command DSTAT

DSTAT

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:

SMGT, operand,[operand],...

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

exit-code	The code that identifies an exit routine to an SMGT operator command is one of the following: <ul style="list-style-type: none"> UEXn where "n" is a single-digit user exit number UXnn where "nn" is a double-digit user exit number HXnn 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 parentheses; 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.

```
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.

```
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.

```
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 OS/390 or z/OS, the dataset ADASNAP must be provided in the Adabas start-up JCL. See the chapter Adabas Session Execution for more information.

```
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 commands are ON/OFF 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.

```
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 command can degrade system performance as long as it is active.

Note:

To use this command successfully under OS/390 or z/OS, the dataset ADASNAP must be provided in the Adabas start-up JCL. See the chapter Adabas Session Execution for more information.

```
SMGT,{ XACTIVATE | XDEACTIVATE }=exit-code
```

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

```
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.

```
SMGT, XLOAD= { exit-code  
                (exit-code, module-name) }
```

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

For more information about the messages displayed using this command, see the Adabas Messages and Codes documentation.

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