



Host Software Component (MVS Implementation)

Reference Summary

Release 5.1

313486602

Proprietary Information Statement

This document and its contents are proprietary to Storage Technology Corporation and may be used only under the terms of the product license or nondisclosure agreement. The information in this document, including any associated software program, may not be reproduced, disclosed or distributed in any manner without the written consent of Storage Technology Corporation.

Limitations on Warranties and Liability

This document neither extends nor creates warranties of any nature, expressed or implied. Storage Technology Corporation cannot accept any responsibility for your use of the information in this document or for your use of any associated software program. You are responsible for backing up your data. You should be careful to ensure that your use of the information complies with all applicable laws, rules, and regulations of the jurisdictions in which it is used.

Warning: No part or portion of this document may be reproduced in any manner or in any form without the written permission of Storage Technology Corporation.

Restrictive Rights

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c) (1) and (2) of the Commercial Computer Software — Restricted Rights at 48 CFR 52.227-19, as applicable.

First Edition, December 2002

Part Number 313486602

EC 128601

This edition applies to Release 5.1 of the Nearline Control Solution (NCS) software. Information in this publication is subject to change. Comments concerning the contents of this publication should be directed to:

Storage Technology Corporation
Manager, Software Information Development
One StorageTek Drive
Louisville, Colorado 80028-5209

or

sid@stortek.com

© 2002 Storage Technology Corporation. All rights reserved. StorageTek, the StorageTek logo and the following are trademarks or registered trademarks of Storage Technology Corporation:

StorageTek®
Automated Cartridge System Library Software (ACSLS)™
Common Library Services (CLS)™
Client System Component (CSC)™
Host Software Component (HSC)™
LibraryStation™
TimberLine™
Silverton™
RedWood™

Other products and names mentioned herein are for identification purposes only and may be trademarks of their respective companies.

Document Effectivity

EC Number	Date	Doc Kit Number	Type	Effectivity
128601	December, 2002	---	First Edition	This document applies to the <i>Host Software Component</i> , Version 5.1.

Contents

About this Summary	ix
Other HSC Books to Help You	ix
Syntax Conventions	1
Syntax Flow Diagrams	1
Specifying Commands	1
Variables	1
Delimiters	1
Flow Lines	2
Single Required Choice	2
Single Optional Choice	3
Defaults	3
Repeat Symbol	3
Syntax Continuation (Fragments)	4
Library Identification	5
How to Specify a CAPid	6
CAPid Formats	6
Ranges and Lists	8
Control Statement Syntax Conventions	11
LIBGEN Macro Syntax	13
SLIACS macro	14
SLIALIST macro	14
SLIDLIST macro	14
SLIDRIVS macro	14
SLIENDGN macro	14
SLILIBRY macro	15
SLILSM macro	16
SLIRCVRY macro	16
SLISTATN macro	16
Utility Syntax	17
ACTivities Report utility	18
AUDIt utility	18
BACKup utility	19
Database Decompile (LIBGEN) utility	19
Directory Rebuild (DIRBLD) utility	19

EJECt utility	19
INITialize Cartridge utility	21
Journal OFFLoad utility	21
MERGEcds utility	21
MOVe utility	22
Reconfiguration utility	22
REPLace utility	22
RESTore utility	22
SCRAtch utility	23
Scratch Conversion utility	23
Scratch Redistribution (SCREdist) utility	24
SET utility	25
UNSCratch utility	26
UNSElect utility	26
Volume Report (VOLRpt) utility	27
Control Statement Syntax	29
Control Data Set Definition (CDSDEF) control statement	30
EXECParm control statement	30
Journal Definition (JRNDEF) control statement	30
License Key Definition (LKEYDEF) command and control statement	31
License Key Information (LKEYINFO) control statement	31
LMUPATH control statement	31
LMUPDEF command and control statement	31
OPTion control statement	32
Reconfiguration Definition (RECDEF) control statement	32
Scratch Subpool Definition (SCRPDEF) command and control statement	32
Scratch Subpool (SCRPOol) control statement	33
Tape Request (TAPEREQ) control statement	34
Tape Request Definition (TREQDEF) command and control statement	36
Unit Attribute (UNITATTR) control statement	36
Unit Attribute Definition (UNITDEF) command and control statement	36
User Exit command and control statement	37
Volume Attribute (VOLATTR) control statement	38
Volume Attribute Definition (VOLDEF) command and control statement	39
HSC Operator Command Syntax	41
Allocation (ALLOC) command and control statement	41
CAP Preference (CAPPref) command and control statement	42
CDs Enable/Disable command	42
CLEan command	42
Communications Path (COMMPath) command and control statement	43
DISMount command	43
Display command	44
DRAin CAP command	49
EJECT command	50

ENter command	51
Journal command	51
MODify command	51
MONITOR command	51
Mount command	52
Mount/Dismount Options (MNTD) command and control statement	53
MOVE command	54
OPTION command and control statement	54
RECover Host command	55
RELEASE CAP command	55
SENter command	55
SRVlev command	55
Stop Monitoring (STOPMN) command	55
SWitch command	56
TRace command	56
TRACELP command	56
User Exit (UEXIT) command and control statement	57
Vary Station command	57
VView command	58
Warn command	59
HSC Diagnostic Command Syntax	61
LIst command	62
TRace command	62

About this Summary

This summary contains frequently used syntax information associated with Release 5.1.0 of the Host Software Component (MVS Implementation).

Use the material presented here as a memory aid. We assume that you are an experienced user who has worked with the Host Software Component at the programmer level. With this in mind, explanatory text has been kept to a minimum.

The following information appears in this reference summary:

- syntax conventions
- LIBGEN macro syntax
- control statement syntax
- utility syntax
- operator command syntax
- diagnostic command syntax.

Other HSC Books to Help You

This summary supplements existing HSC 5.1.0 documentation. For more detailed information about a topic, refer to the following manuals:

- *HSC Configuration Guide*
- *HSC System Programmer's Guide*
- *HSC Operator's Guide*.

Syntax Conventions

Syntax Flow Diagrams

Syntax is illustrated using flow diagrams. These can include the following elements:

- Syntax – the diagram itself.
- Items – individual elements inside the diagram. Items can be keywords, variables, delimiters, operators, fragment references, and separators.
- Groups – a collection of items or other groups.

The following sections describe syntax flow diagram features and include some generic examples.

Specifying Commands

Commands are composed of command names, keyword parameters, and positional parameters. Command names initiate command execution, keyword parameters are operands that contain keywords and their related values, and positional parameters are operands that are identified by their position in the command string rather than by keywords.

- Keyword parameters can be specified in any order. The HSC accepts (tolerates) multiple occurrences of a keyword. The value assigned to a keyword reflects the last occurrence of a keyword within a command.
- Positional parameters must be entered in the order shown in the syntax diagram.
- Uppercase letters indicate the minimum abbreviation for the command name, keyword, or positional parameter.

Variables

Variables are italicized.

Delimiters

If a comma(,), a semicolon(;), or other delimiter is shown with an element of the syntax diagram, it must be entered as part of the statement or command.

Flow Lines

Syntax diagrams consist of horizontal and vertical lines and the text of a command, control statement, macro, or utility.

►►—COMMAND/MACRO/UTILITY————→

or

►►—Item1————→
|————Item2————→
|————Item3————→

Diagrams are read left to right and top to bottom. Arrows show flow and direction.

- a statement begins with ►►
- a statement ends with ►◄
- diagrams continuing to the next line begin with ►
- fragments begin and end with |

►►—COMMAND/UTILITY NAME——Item1(variable1)——Item2(variable2)————→
|————variable3————→
|————variable4————→
————→ Item3(variable5)————→

Single Required Choice

Branch lines, without repeat arrows, indicate that a **single** choice must be made. If one of the items from which a choice is being made is on the base line of the diagram, a single choice is required.

►►—Item1————→
|————Item2————→
|————Item3————→

Single Optional Choice

If the first item is on the line below the base line, a single choice of items in the stack is optional.



Defaults

Default values and parameters appear above the syntax diagram line. In the following example, if a value is not specified with the command, Default Value is used by the HSC.

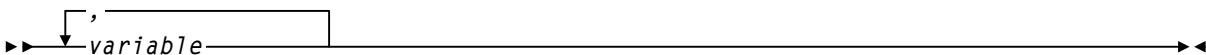


Some keyword parameters provide a choice of values in a stack. When the stack contains a default value, the keyword and the value choices are placed below the baseline to indicate that they are optional, and the default value appears above the keyword line. In the following example, if the keyword is not specified with the command, Keyword(Default Value) is used by the HSC.



Repeat Symbol

A repeat symbol indicates that more than one choice can be made or that a single choice can be made more than once. The repeat symbol shown in this example indicates that a comma is required as the repeat separator.



Syntax Continuation (Fragments)

Fragment references direct you to parts (fragments) of the syntax that contain more detail than can be shown in the main syntax diagram.

►►COMMAND/UTILITY NAME——| Fragment Reference |————►

Fragment:

|—Item1(—variable1—, —variable3—)
|—variable2 |—variable4|————►

►—Item2(—variable5—, —variable7—)
|—variable6 |—variable8|————►

Library Identification

Each ACS, LSM, and CAP is assigned a unique identification number during LIBGEN. Use this number in HSC commands and utilities when identifying a specific ACSid, LSMid, or CAPid.

- ACSid (*acs-id*) is a hexadecimal value from 00 through FF that identifies the LMU.

An *acs-id* is the result of defining the SLIALIST macro during a LIBGEN. See “LIBGEN Macros” in the *HSC Configuration Guide* for information about the SLIALIST macro. The first ACS listed in this macro acquires a hexadecimal identifier of 00, the second ACS listed acquires a hexadecimal identifier of 01, and so forth until all ACSs are identified.

- An LSM number (*l*) is a hexadecimal value from 0 through F. It differentiates an LSM from every other LSM connected to the same LMU.

An LSM number is the result of defining the SLIACS macro LSM parameter. See “LIBGEN Macros” in the *HSC Configuration Guide* for information about the SLIACS macro. The first LSM listed for an ACS acquires a hexadecimal number of 0, the second LSM listed for an ACS acquires a hexadecimal number of 1, and so forth.

- An LSMid (*lsm-id*) is the concatenation of the ACSid and the LSM number. It differentiates an LSM from every other LSM in a library.
- A CAP number is a hexadecimal value from 00 to 02. The CAP number identifies a specific CAP in an LSM that has more than one CAP.
- A CAPid (*cap-id*) is a hexadecimal value made up of the LSMid and the CAP number separated by a colon. Refer to “How to Specify a CAPid” on page 6 for more information.

Some HSC commands and utilities require, or optionally allow, the user to specify a host identifier or a VOLSER.

- The *host-id* for a given host is the identifier specified in the HOSTID parameter of the SLILIBRY macro in the LIBGEN: the SMF system identifier for JES2, or the main processor name for JES3. Valid characters for a HOSTID are A-Z, 0-9, #, \$ and @.
- A VOLSER (*volser*) identifies a volume serial number consisting of one to six characters. Valid characters are A-Z, 0-9, # (crosshatch), \$, ¥ (yen character), and optional trailing blanks. Leading blanks are not allowed.

How to Specify a CAPid

A CAPid specifies a particular CAP in the library. Each CAP is identified by the LSMid of the LSM that the CAP is attached to and a CAP number to distinguish it from other CAPs in that LSM.

CAP configurations differ based on the LSM type. The following configurations are possible:

LSM (Model 4410) and PowderHorn LSM (Model 9310)

can be configured with either the standard 21-cell CAP or an enhanced CAP. An enhanced CAP contains two 40-cell magazine-style CAPs and a one-cell priority CAP (PCAP). The 40-cell CAPs function independently.

WolfCreek LSM (Models 9360-050, 9360-075, and 9360-100)

configured with a WolfCreek CAP which contains a 20-cell magazine-style CAP and a PCAP. An optional 30-cell, magazine-style CAP, called a WolfCreek optional CAP, may be added to the WolfCreek CAP.

TimberWolf LSM (Model 9740)

configured with either a 14-cell permanent rack or a 10-cell removable magazine.

CAPid Formats

There are two formats that can be used to specify a CAPid:

- AAL , where AA is the ACSid and L is the LSM number. This format is referred to as the *lsm-id*.
- $AAL:CC$, where AAL is the LSMid and $:CC$ is the CAP number. This format is referred to as the *cap-id*.



Caution: Do not use a colon as the MVS command delimiter because the system will process the colon in the new CAPid format as the end of the command. All information following the colon in an HSC command will not be processed.

The appropriate format to use is determined by the CAP hardware and the command being specified.

- The AAL format can be specified in the following situations:
 - to specify a standard CAP
 - to allow the HSC to select a CAP based on CAP preference.

- The *AAL:CC* format can be specified for any CAP hardware in any command that accepts a CAPid. This is the preferred format. Valid CAP numbers are:

00 indicates one of the following:

- a 21-cell standard CAP
- the right-hand 40-cell CAP of an enhanced CAP
- the 20-cell CAP of a WolfCreek CAP
- a 14-cell or 10-cell removable magazine 9740 TimberWolf CAP.

01 indicates one of the following:

- the left-hand 40-cell CAP of an enhanced CAP
- the 30-cell CAP of a WolfCreek optional CAP.

02 indicates the PCAP in either an enhanced CAP or a WolfCreek CAP.

Ranges and Lists

HSC commands and utilities often allow the user to specify ranges and lists of elements.

1. An inclusive range is indicated by a pair of elements of the same length and data type, joined by a dash. The first element **must** be strictly less than the second element.
 - A hexadecimal range consists of a pair of hexadecimal numbers (for example, 0A2-0AD, or 000-0FC).
 - A decimal range consists of a pair of decimal numbers (for example, 1-9, or 010-094). Leading zeros are not required.
 - A numeric VOLSER range (*vol-range*) consists of a pair of VOLSER elements containing a decimal numeric portion of 1 to 6 digits (for example, ABC012-ABC025, or X123CB-X277CB). The decimal portion is referred to as an incremental range. The following additional restrictions apply:
 - The character positions of the incremental portion of both range elements must match.
 - The non-incremental characters of the first element must be identical to those of the second element.
 - You cannot increment two portions of a range element. If 111AAA is the first element, you cannot specify 112AAB for the second element.
 - If a VOLSER range contains more than one decimal portion, only the right-most portion is valid as the incremental range. For example:

A00B00 the largest range that can be specified is A00B00 through A00B99.

A0B0CC the largest range that can be specified is A0B0CC through A0B9CC.

000XXX the largest range that can be specified is 000XXX through 999XXX.



Note: A VOLSER range for most operator commands is limited to 100 entries. If a larger range is entered, only the first 100 VOLSERs in the range are acted on. If HSC utilities are used, the entire range is processed.

- An alphabetic VOLSER range (*vol-range*) consists of a pair of VOLSER elements containing an incremental portion of 1 to 6 characters (for example, 000AAA-000ZZZ, or 9AAA55-9ZZZ55). This portion is referred to as an incremental range. The following additional restrictions apply:
 - The character positions of the incremental portion of both range elements must match.
 - The non-incremental characters of the first element must be identical to those of the second element.

- You cannot increment two portions of a range element. If 111AAA is the first element, you cannot specify 112AAB for the second element.
- The alphabetic portion of the VOLSER range is defined as being from character A to Z. To increment multi-character sequences, each character increments to Z. For instance, ACZ is part of the AAA-AMM range. Examples are:

A<u>00A0-A99A0</u>	increments VOLSERs A00A0 through A09A0, then A10A0 through A99A0.
9<u>AA9A-9ZZ9A</u>	increments VOLSERs 9AA9A through 9AZ9A, then 9BA9A through 9ZZ9A.
111<u>AAA-111ZZZ</u>	increments VOLSERs 111AAA through 111AAZ, then 111ABA through 111ZZZ.
999<u>AM8- 999CM8</u>	increments VOLSERs 999AM8 through 999AZ8, then 999BA8 through 999CM8
A3<u>BZZ9- A3CDE9</u>	increments VOLSERs A3BZZ9 through A3CAA9, then A3CAB9 through A3CDE9
<u>AAAAAA-</u> <u>AAACCC</u>	increments VOLSERs AAAAAA through AAAAAZ, then AAAABA through AAACCC
<u>CCCNNN-</u> <u>DDDNNN</u>	increments VOLSERs CCCNNN through CCCNNZ, then CCCNOA through DDDNNN [*]

* **Caution:** This is a very large range.

The number of volumes in an alphabetic VOLSER range depends on the number of elements in the incrementing portion of the VOLSER range. For an A to Z range in each character position, the number of volumes can be calculated by 26 to the power of the number of positions that are being incremented.

A-Z	26^1	26
AA-ZZ	26^2	676
AAA-ZZZ	26^3	17,576
AAAA-ZZZZ	26^4	456,976
AAAAAA-ZZZZZ	26^5	11,881,376
AAAAAAA-ZZZZZZ	26^6	308,915,776



Note: For most operator commands, a VOLSER range is limited to 100 entries. If a large range is entered, only the first 100 VOLSERs are acted upon. If HSC utilities are used, the entire range is processed.

2. A list consists of one or more elements. If more than one element is specified, the elements **must** be separated by a comma or a blank, and the entire list enclosed in parentheses.

- For some HSC operator commands, an element may consist of a single item or a range. Refer to the individual command explanations for valid list entries.
- In general, HSC utilities **do not allow** ranges to be specified in a list. The exception to this is a VOLSER list (*vol-list*) which does allow ranges to be specified.

For VOLATTR control statements, you can use wildcard characters (%,, ?, or *) to identify a list of VOLSERs.

Control Statement Syntax Conventions

The control statement for each utility program consists of a command (indicating the utility function) followed by parameters, as applicable, in 80-character card-image records. The standard syntax conventions for control statements are as follows:

- The only valid control statement information area is from column 2 to column 72. Columns 73-80 are ignored.
- Parameters are separated by one or more blanks or a comma.
- A value is associated with a parameter by an equal sign (=) or by enclosing the value in parentheses, and concatenating it immediately after the parameter.
- Case (upper or lower) is ignored in actual control statements.
- Control statements may be interspersed with comments designated by an asterisk (*) in column one.

To allow for continuation, comments in the job stream must start with /* and end with */. Comments cannot be nested, and mixing the two comment styles (* and /*) is not allowed.

For definition data sets (VOLATTRs, UNITATTRs and TAPEREQs) comments **must** be in the new format /*...*/. Asterisk (*) comments are not allowed. A /*...*/ comment in the first line is **not** required for definition data sets.

A control statement is terminated if the statement is not continued. Control statements must have a /*...*/ comment as the **first** control statement in the PARMLIB member. A PARMLIB member that does not begin with a /*...*/ style comment is assumed to be in the old format. Comments in old format members must begin with an asterisk in column 1.

In contrast to utility control statements, PARMLIB control statements may begin in column 1. Columns 73-80 are ignored.

- The 80-character card-image records use conventional continuation rules.
 - A space and a dash (-) following a parameter or parameter value indicates that a blank is to be inserted between the last nonblank character of this line and the first nonblank character of the next nonblank record.
 - A plus sign (+) specifies that the continued control information is to be concatenated directly after the character preceding the plus sign. The continued data starts at column two of the next nonblank record.

Note: You can use a continuation only after a new keyword or after the natural end of a value. Some examples follow.

The following examples illustrate continuations used correctly:

```
SCRPOOL NAME=STD36, RANGE+
(AAA000-AAA999, ZZZ000-ZZZ999)
```

```
SCRPOOL NAME=STD36, RANGE(AAA000-AAA999, -
ZZZ000-ZZZ999)
```

The following example illustrates a continuation used incorrectly:

```
SCRPOOL NAME=STD36, RANGE(AAA000-AAA999, ZZZ+
000-ZZZ999)
```

- Users must enter a nonblank character in column 72 (e.g., an X).
- PARMLIB control statements can be continued using the preceding continuation rules **only** if they are new format control statements.
- The maximum length of a control statement is 32,767 characters.
- The maximum length of a command (used as a command or in PARMLIB) is 126 characters.

LIBGEN Macro Syntax

This section contains the syntax for LIBGEN macros presented in alphabetical order. For complete descriptions of the macros and the required order for specifying them, see the *HSC Configuration Guide*.

SLIACS macro

```
►►acs0—SLIACS— Additional Parameters →  
  
Additional Parameters:  
|—ACSDRV=(esoteric0, ...,esoteric15)→  
|—[ ,STATION=(station0, ...,station15)] ,LSM=(lsm0,lsm1, ...,lsm15)→
```

SLIALIST macro

```
►►acslist—SLIALIST—acs0, acs1,....acs255→
```

SLIDLIST macro

```
►►drvelst0—SLIDLIST—HOSTDRV=(drives0,...,drives15)→
```

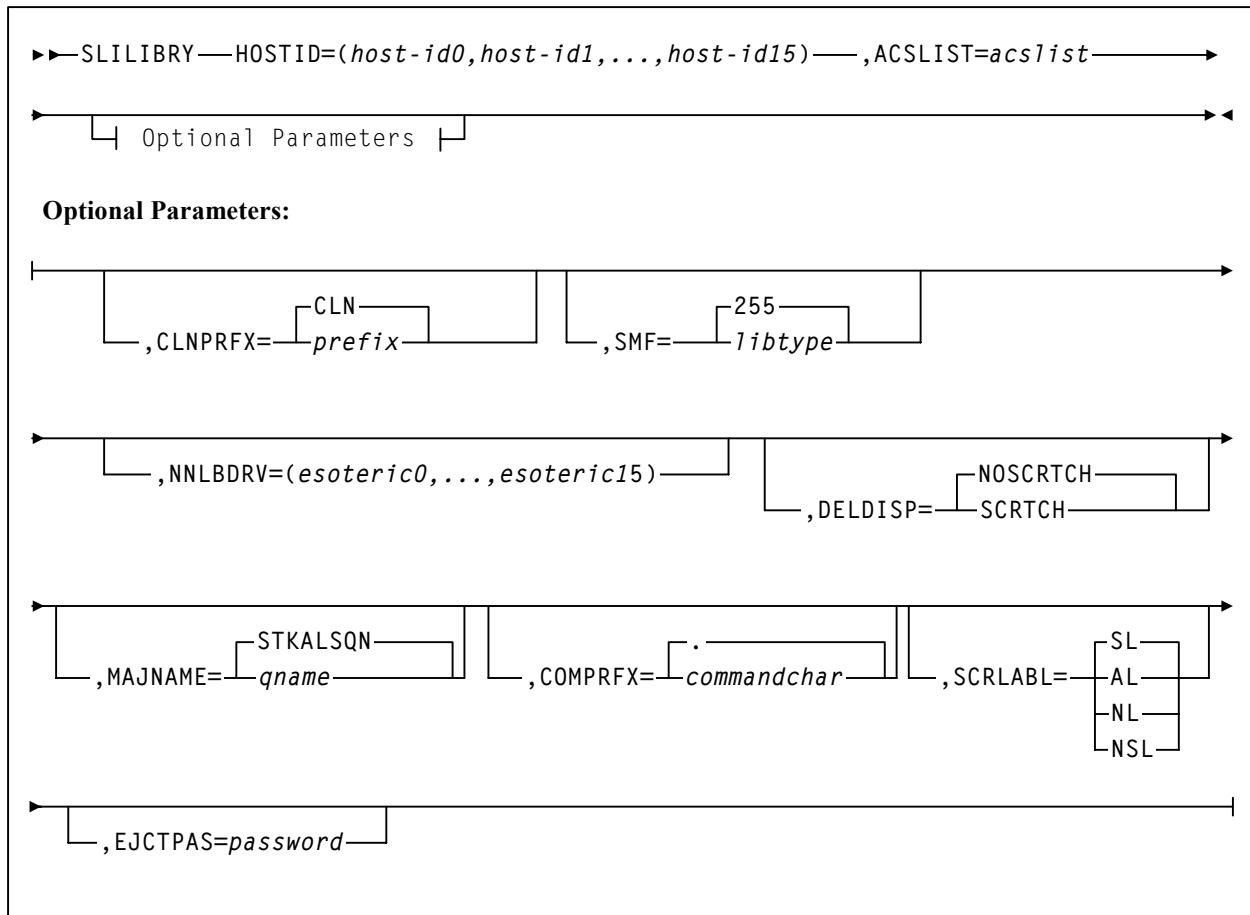
SLIDRIVS macro

```
►►drives0 SLIDRIVS ADDRESS=(addr0,addr1...)→
```

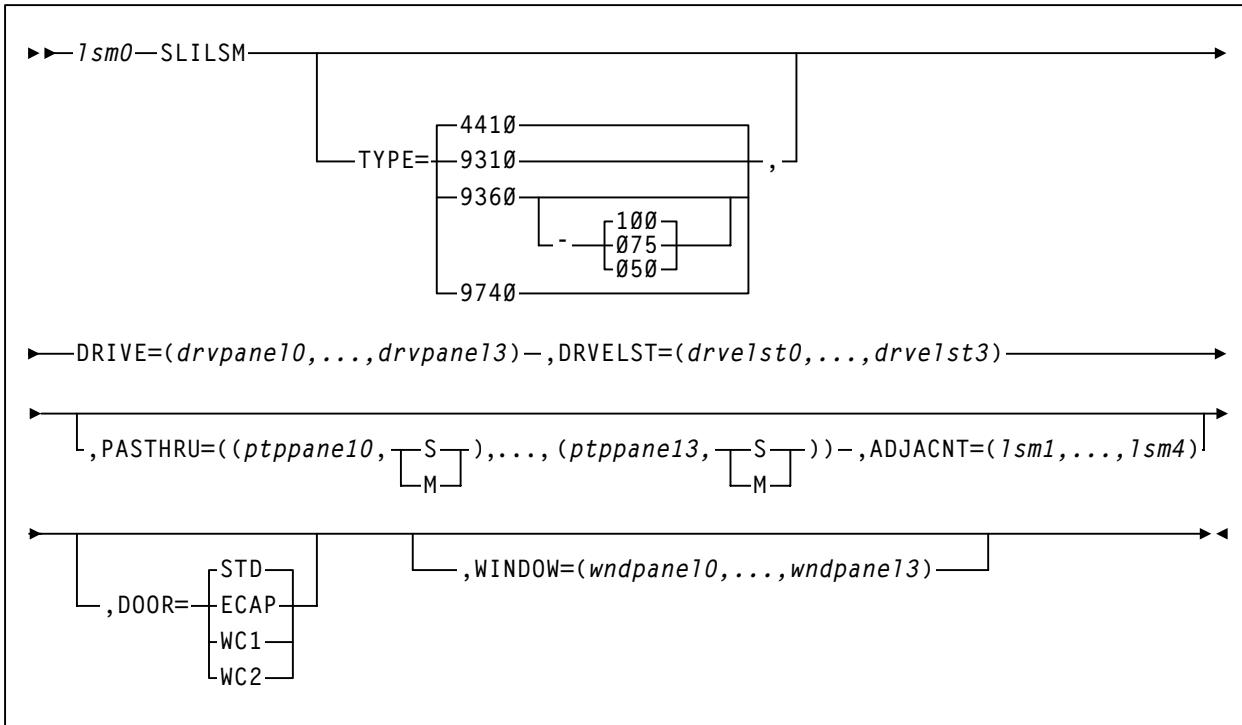
SLIENDGN macro

```
►►SLIENDGN→
```

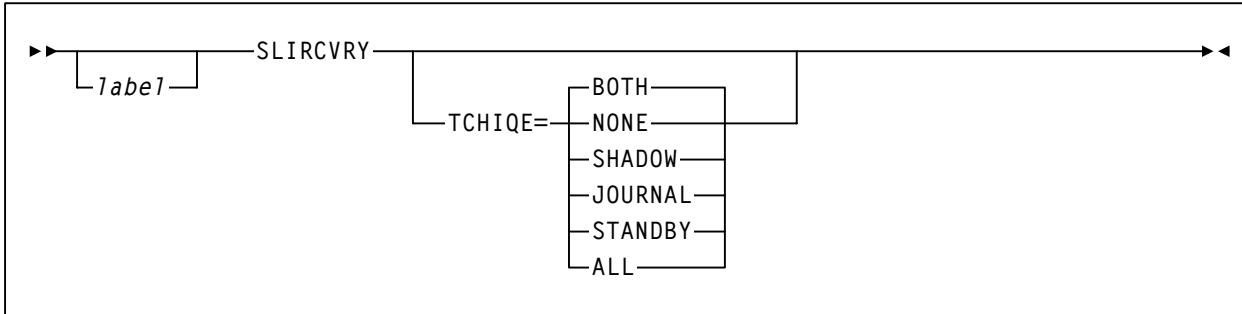
SLILIBRY macro



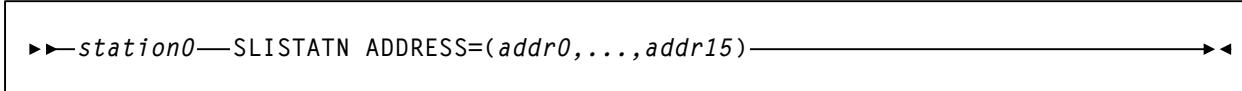
SLILSM macro



SLIRCVRY macro



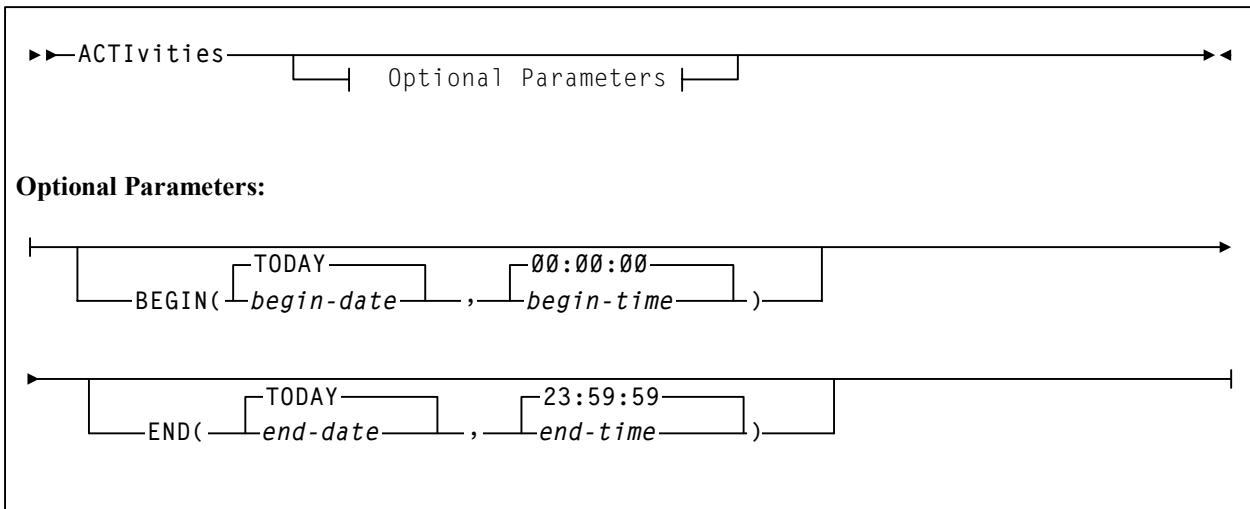
SLISTATN macro



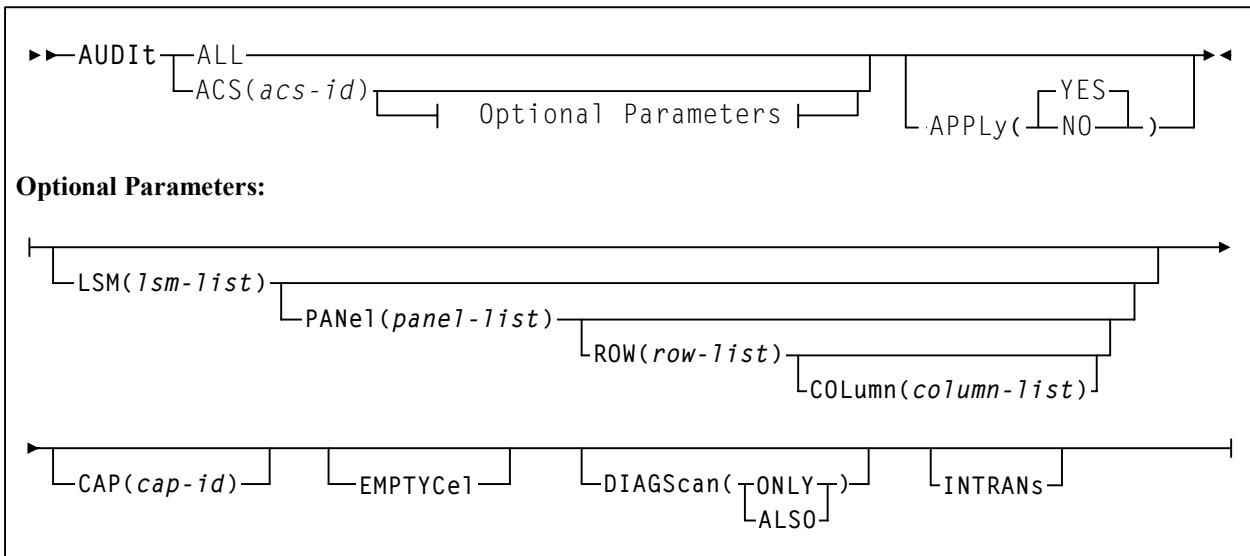
Utility Syntax

This section contains the syntax for library utilities. For complete descriptions of the utilities, see the *HSC System Programmer's Guide*.

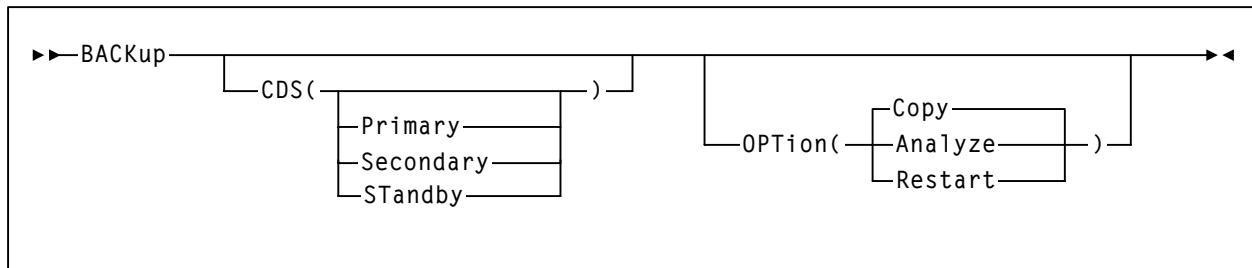
ACTIvities Report utility



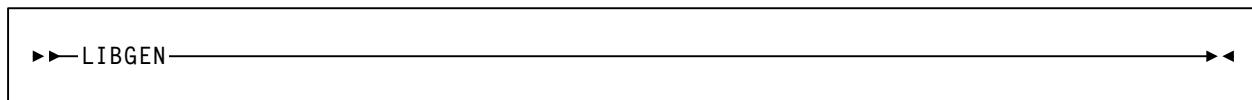
AUDIt utility



BACKup utility



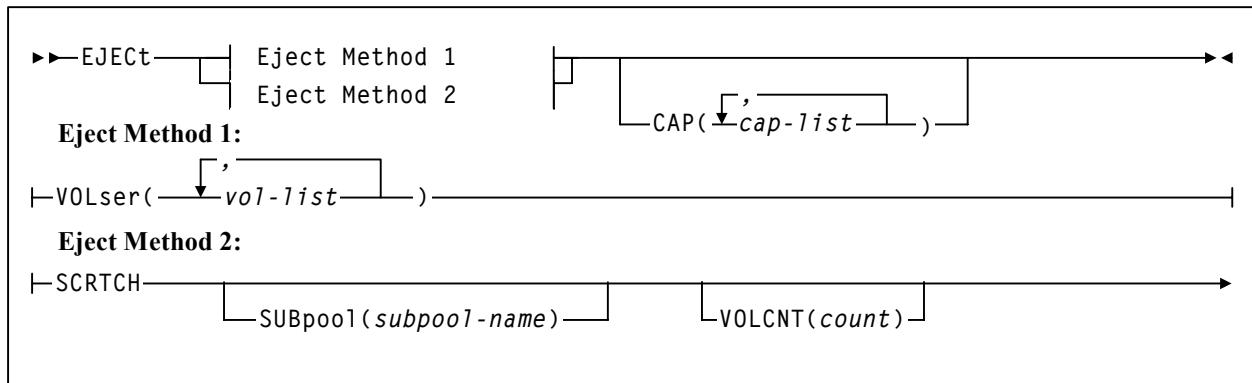
Database Decompile (LIBGEN) utility



Directory Rebuild (DIRBLD) utility



EJECT utility

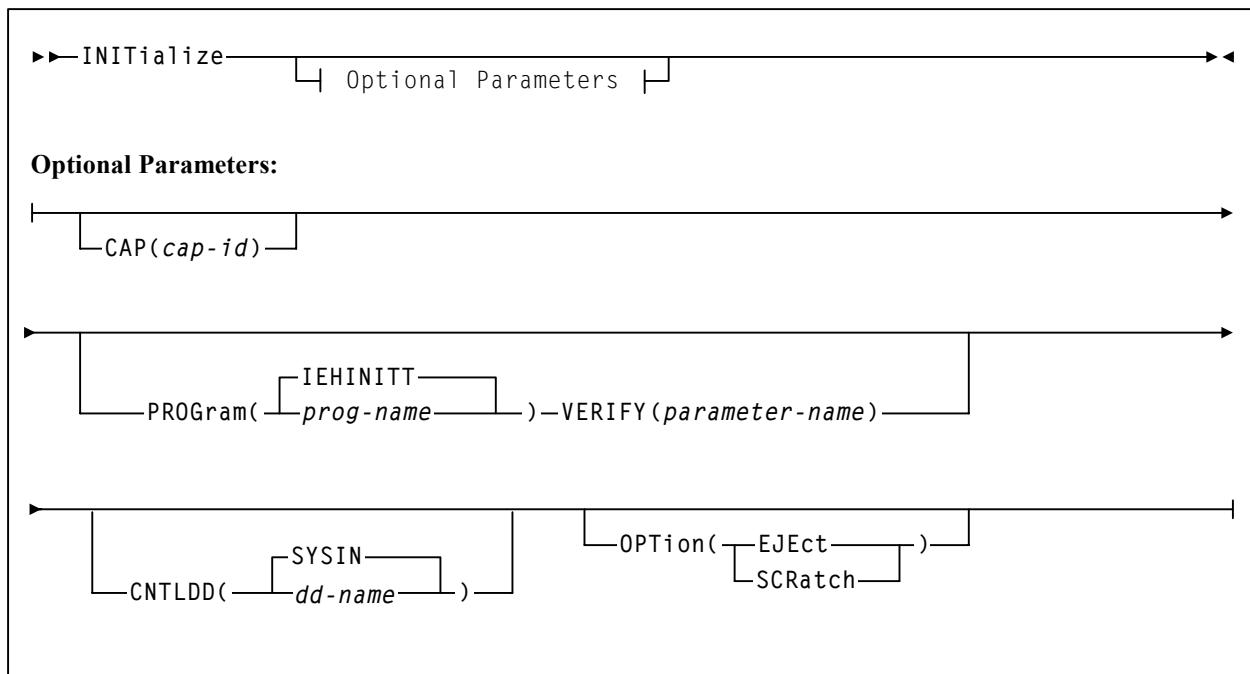


EJECt utility (continued)

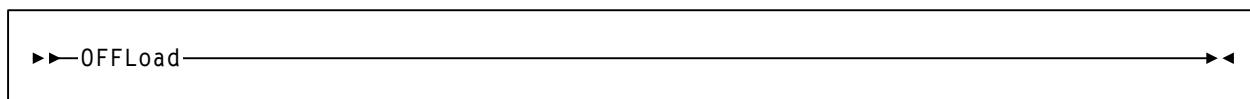
Eject Method 2 (continued):

MEDIA(Standard)
-CST		
-MEDIA1		
-STD		
-1		
-348Ø		
-ECART		
-E		
-ECCST		
-ETAPE		
-Long		
-MEDIA2		
-349ØE		
-ZCART		
-Z		
-DD3A		
-DD3B		
-DD3C		
-STK1		
-STK1R		
-R		
-STK2		
-STK2P		
RECtech(18track)
	-36Atrack	
	-36Btrack	
	-36Ctrack	
	-DD3	
	-STK1R	
	-STK1R34	
	-STK1R35	
	-STK1RA	
	-STK1RA34	
	-STK1RA35	
	-STK1RB	
	-STK1RB34	
	-STK1RB35	
	-STK2P	
	-STK2P34	
	-STK2P35	
	-STK2PB	
	-STK2PB34	
	-STK2PB35	

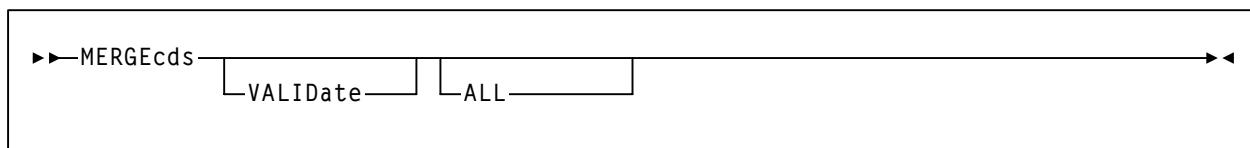
INITialize Cartridge utility



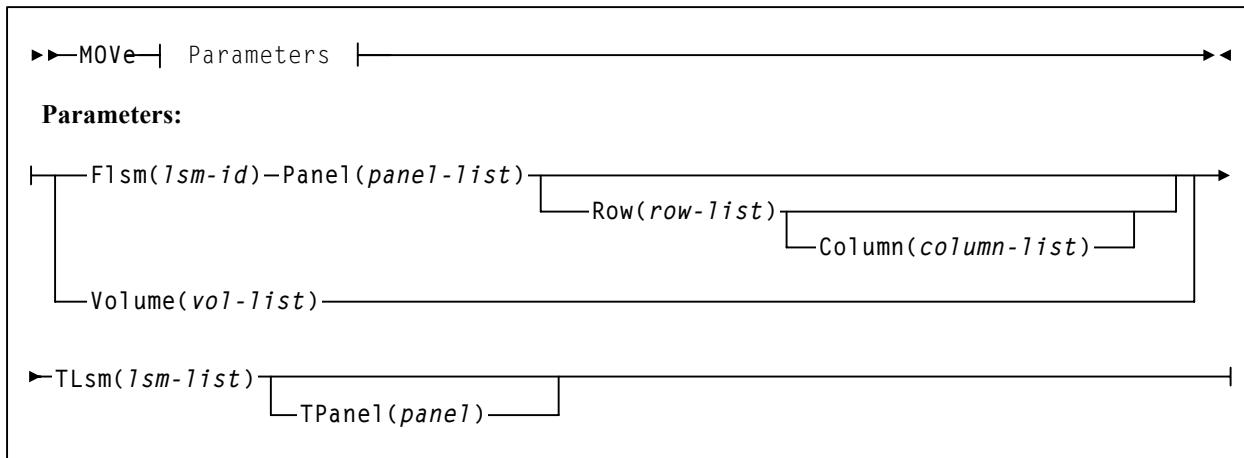
Journal OFFLoad utility



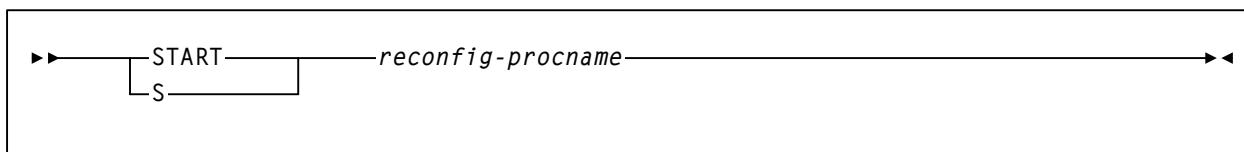
MERGEcds utility



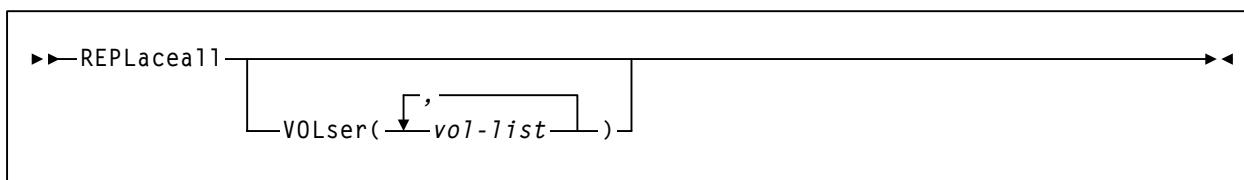
MOVe utility



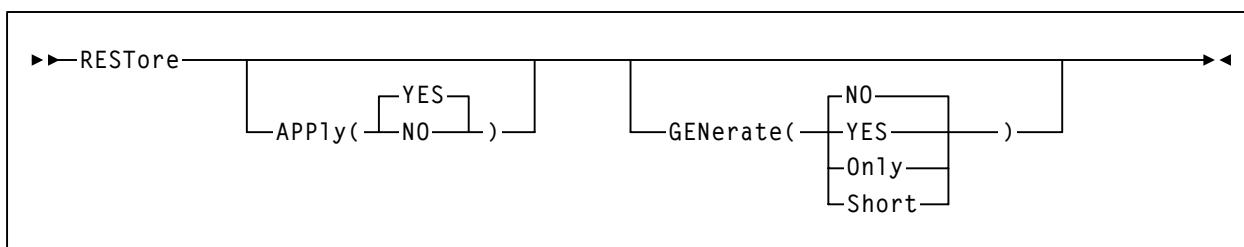
Reconfiguration utility



REPLace utility



RESTore utility



SCRAtch utility

```
►►SCRAtch—VOLser( [ vol-list ] )
```

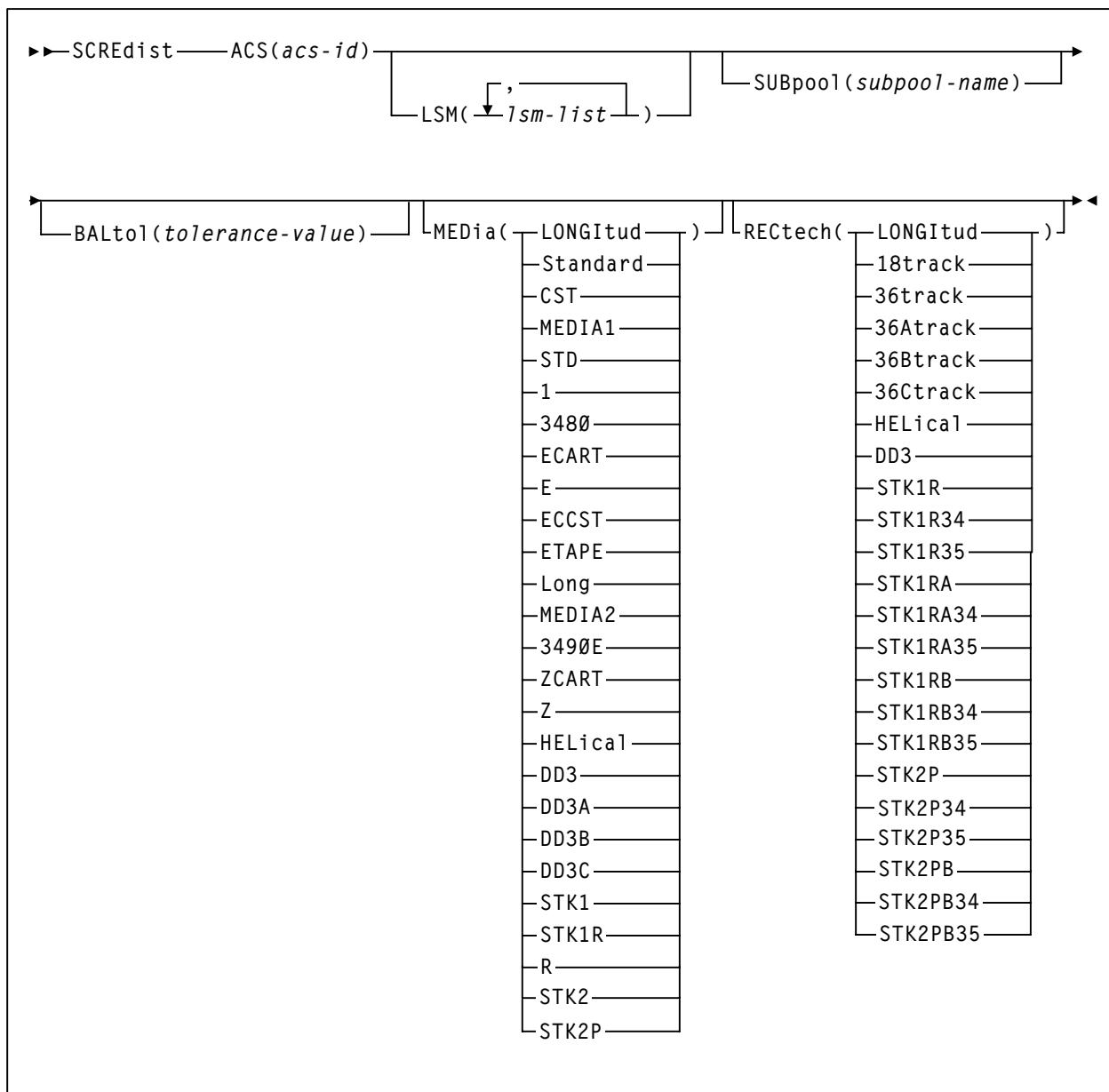
Scratch Conversion utility

```
►►PARM= ' [ Optional Parameters ] '
```

Optional Parameters:

```
[ TMS ] , [ TLMS ] , [ RMM ] , SCRPOOL( [ SL ] , [ NL ] , [ NSL ] , [ AL ] , [ SUL ] ) , [ 00001 ] , [ 1900001 ] , [ TODAY ] , [ yyddd1 ] - [ yyddd2 ] , [ yyyyddd1 ] - [ yyyyddd2 ]  
[ MIXED ] , [ LIBONLY ]
```

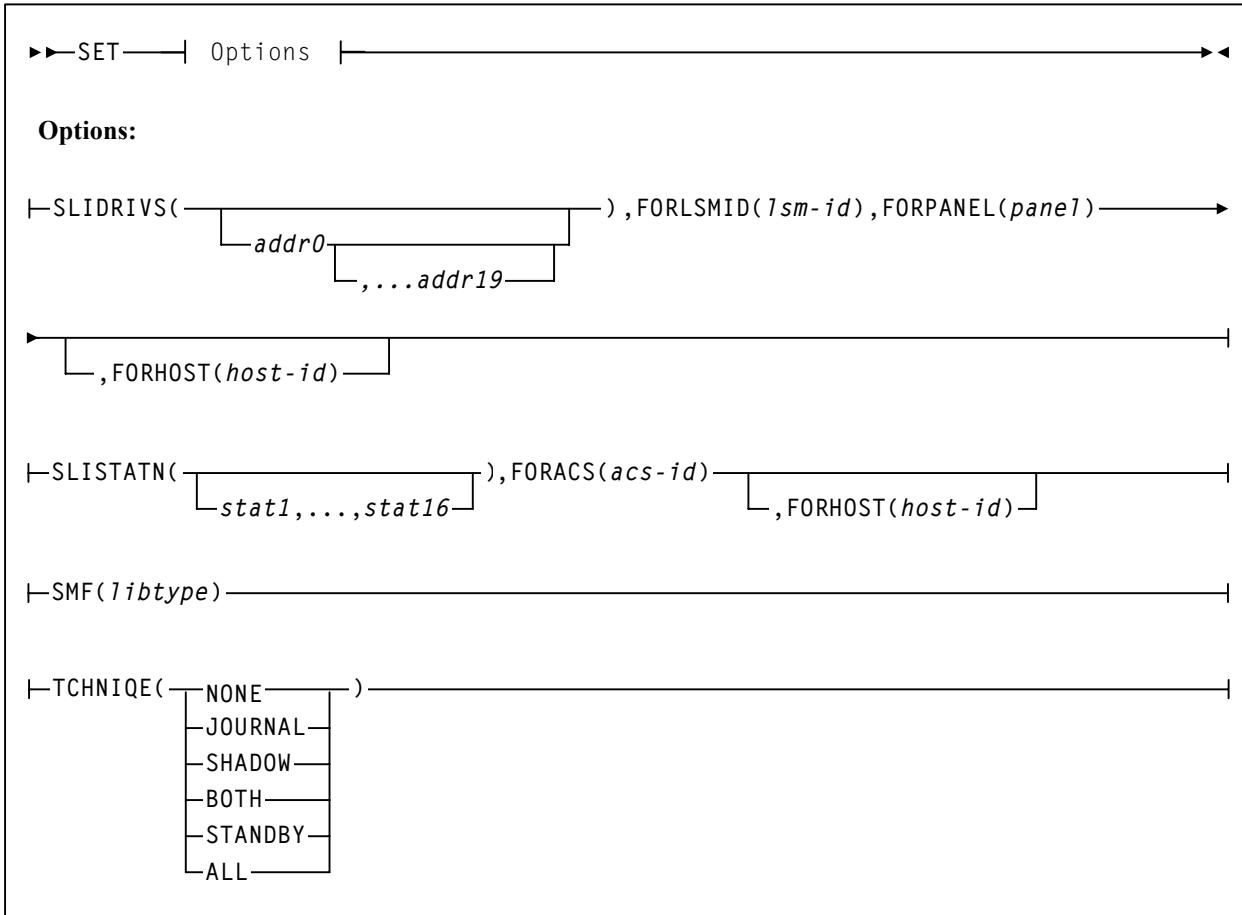
Scratch Redistribution (SCREdist) utility



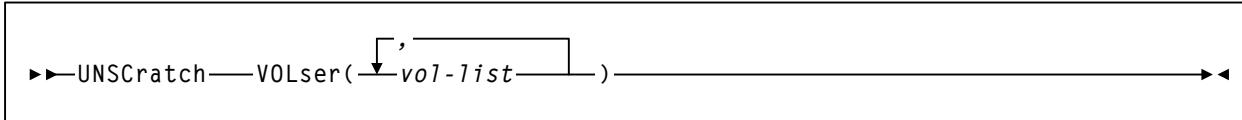
SET utility

```
►►SET— Options →  
  
Options:  
|—ACSDRV(esoteric)—,FORACS(acs-id)— [ ,FORHOST(host-id) ]  
|—CDSLVL [ CONVERT(210) ] [ REGRESS(200) ]  
|—CLNPRFX(prefix)—  
|—COMPRFX(cmdhex)—  
|—DELDISP( [ SCRTCH ] [ NOSCRTCH ] )—  
|—EJCTPAS( [ newpswd ] [ ,OLDPASS(oldpswd) ] )—  
|—FREEZE( [ ON ] [ OFF ] ),FORLSMID(lsm-id),FORPANEL(panel)—  
|—HOSTID(newhost),FORHOST(oldhost)—  
|—HSCLEVEL(OFF),FORHOST(host-id)—  
|—MAJNAME(qname)—  
|—NEWHOST(newhost),LIKEHOST(model-host)—  
|—NNLBDRV( [ esoteric ] [ ,FORHOST(host-id) ] )—  
|—SCRLABL( [ SL ] [ AL ] [ NL ] [ NSL ] )—
```

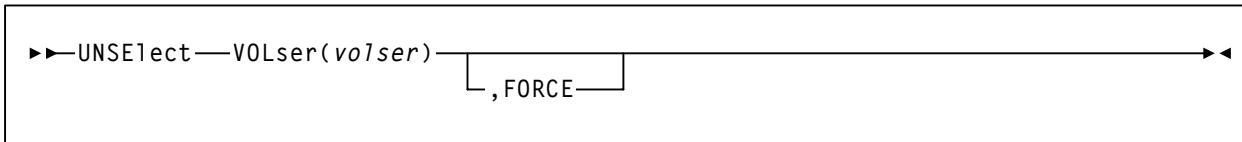
SET utility (continued)



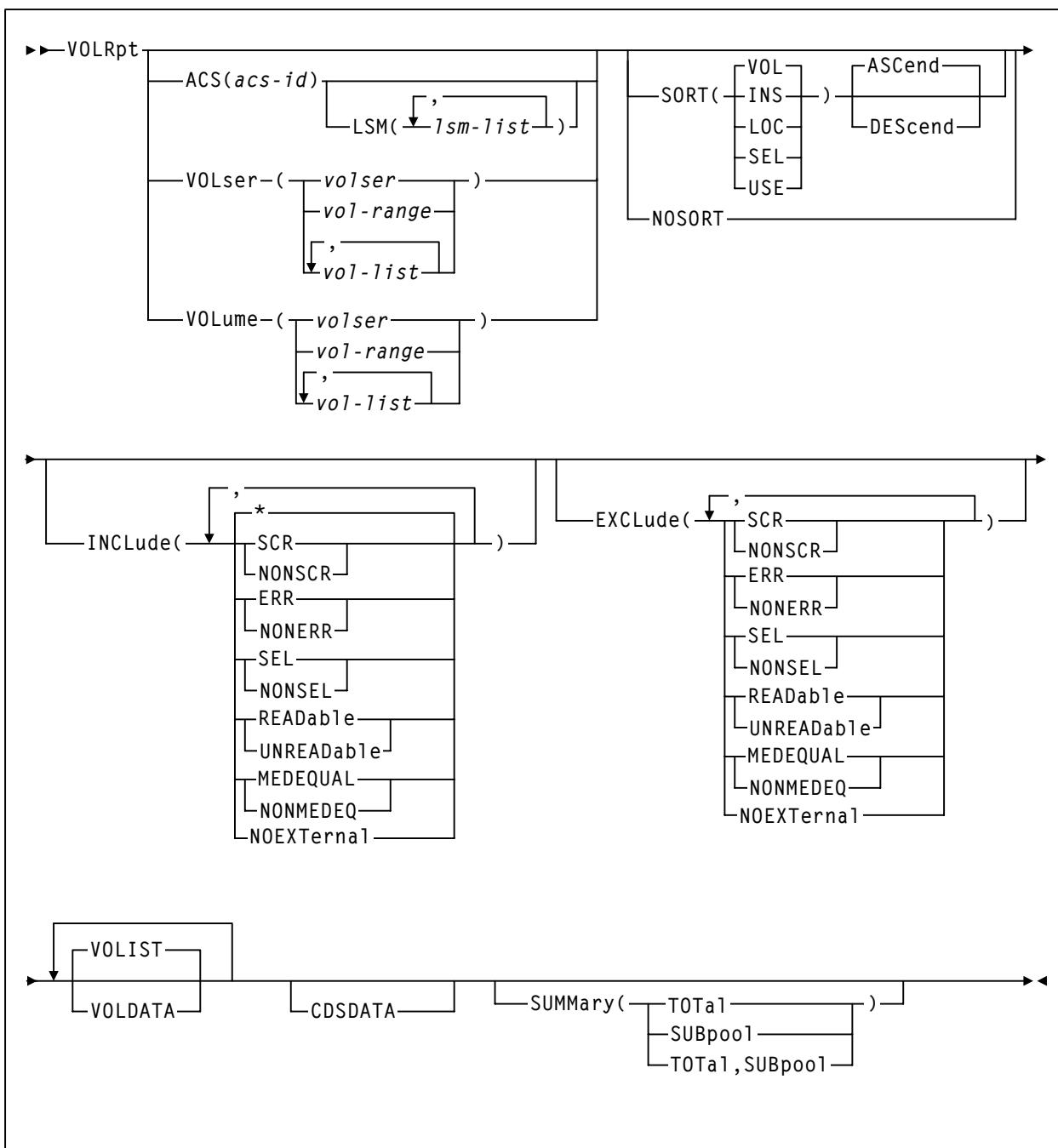
UNSCratch utility



UNSelect utility



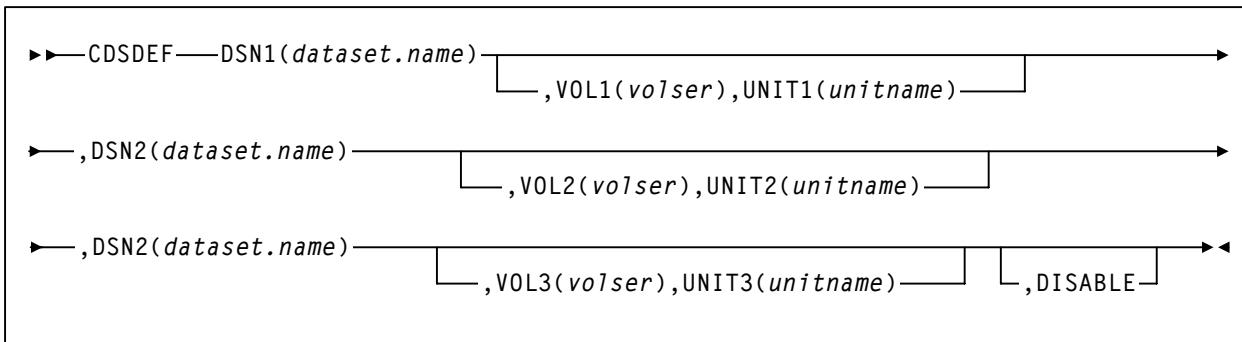
Volume Report (VOLRpt) utility



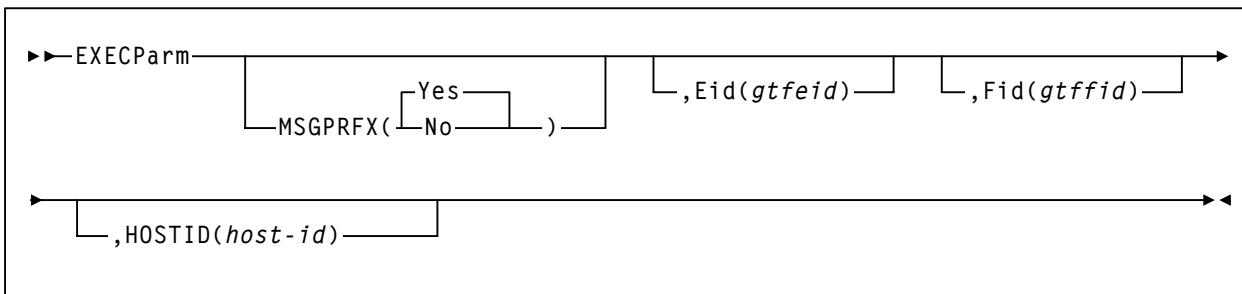
Control Statement Syntax

This section contains the syntax for control statements. For complete descriptions of the control statements, see the *HSC System Programmer's Guide*.

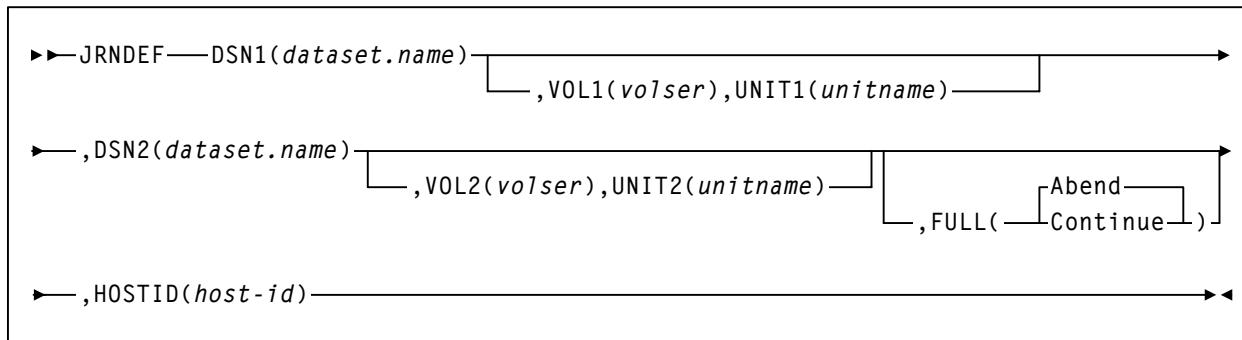
Control Data Set Definition (CDSDEF) control statement



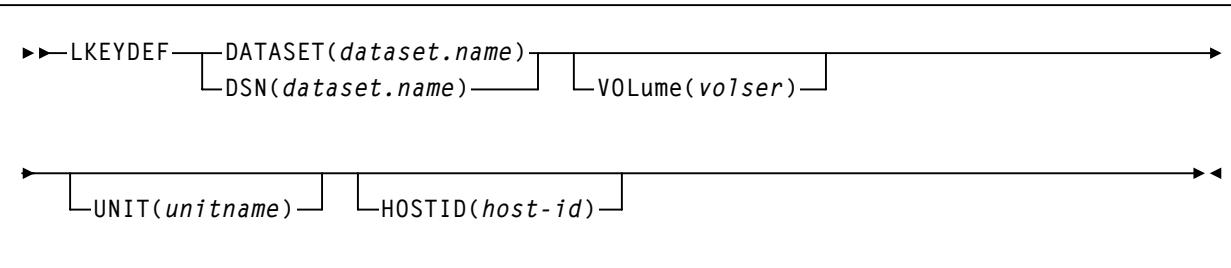
EXCParm control statement



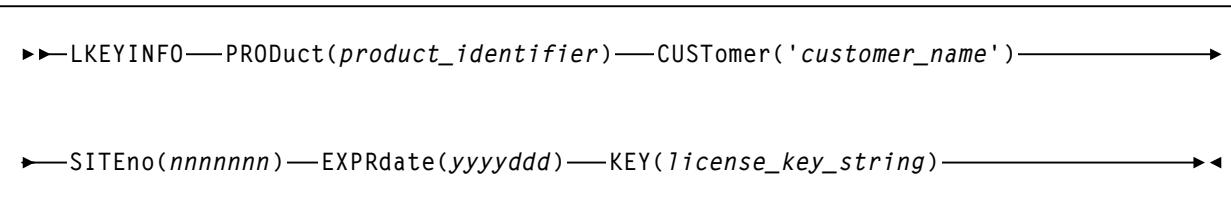
Journal Definition (JRNDEF) control statement



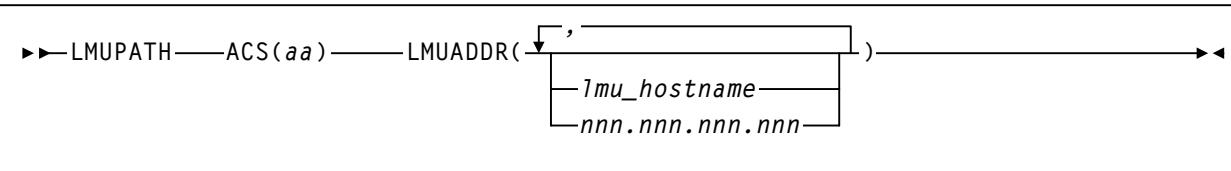
License Key Definition (LKEYDEF) command and control statement



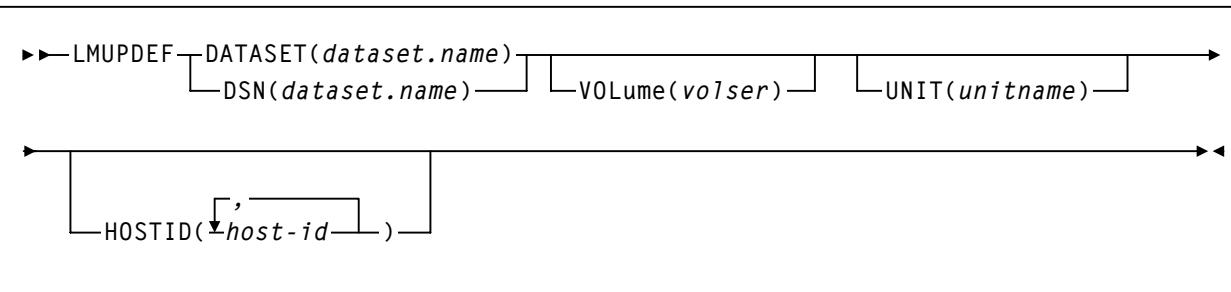
License Key Information (LKEYINFO) control statement



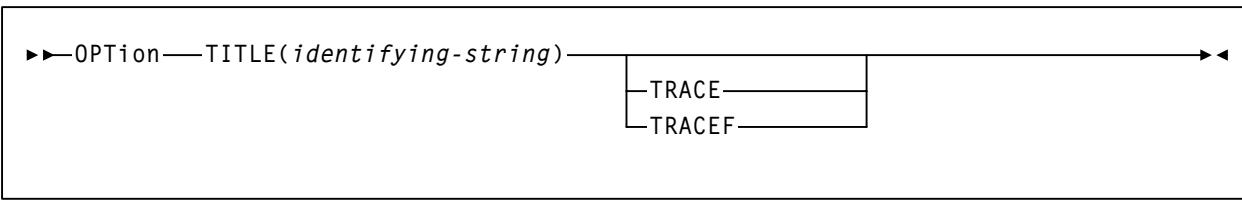
LMUPATH control statement



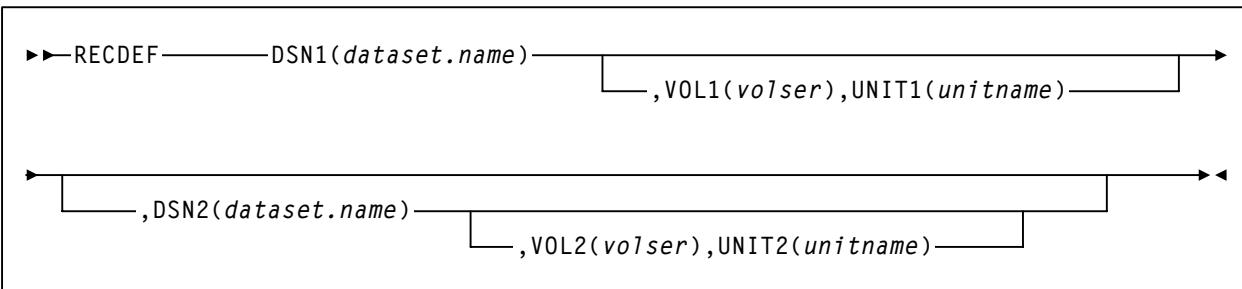
LMUPDEF command and control statement



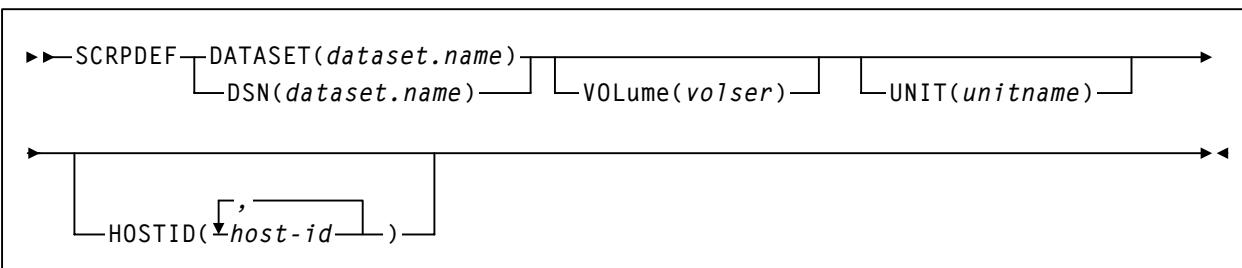
OPTION control statement



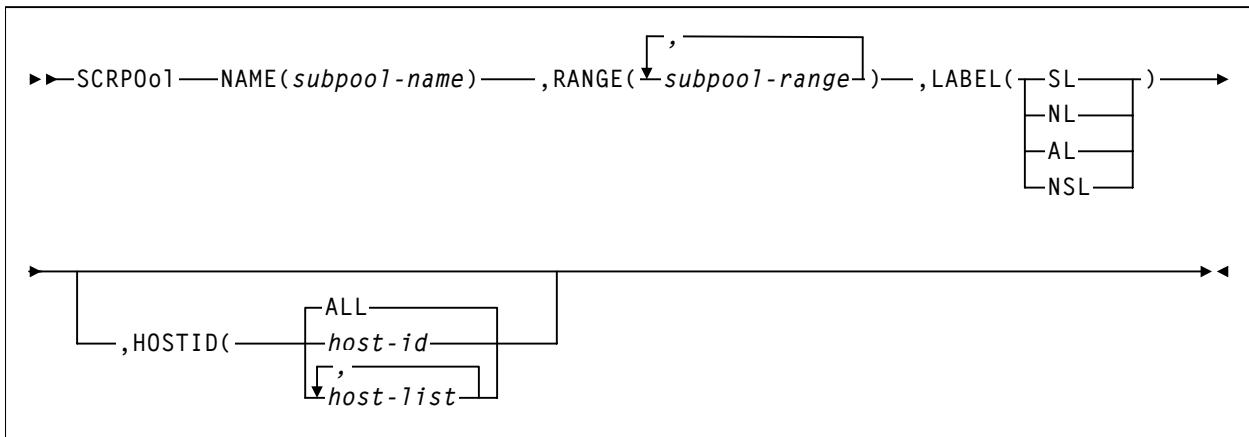
Reconfiguration Definition (RECDEF) control statement



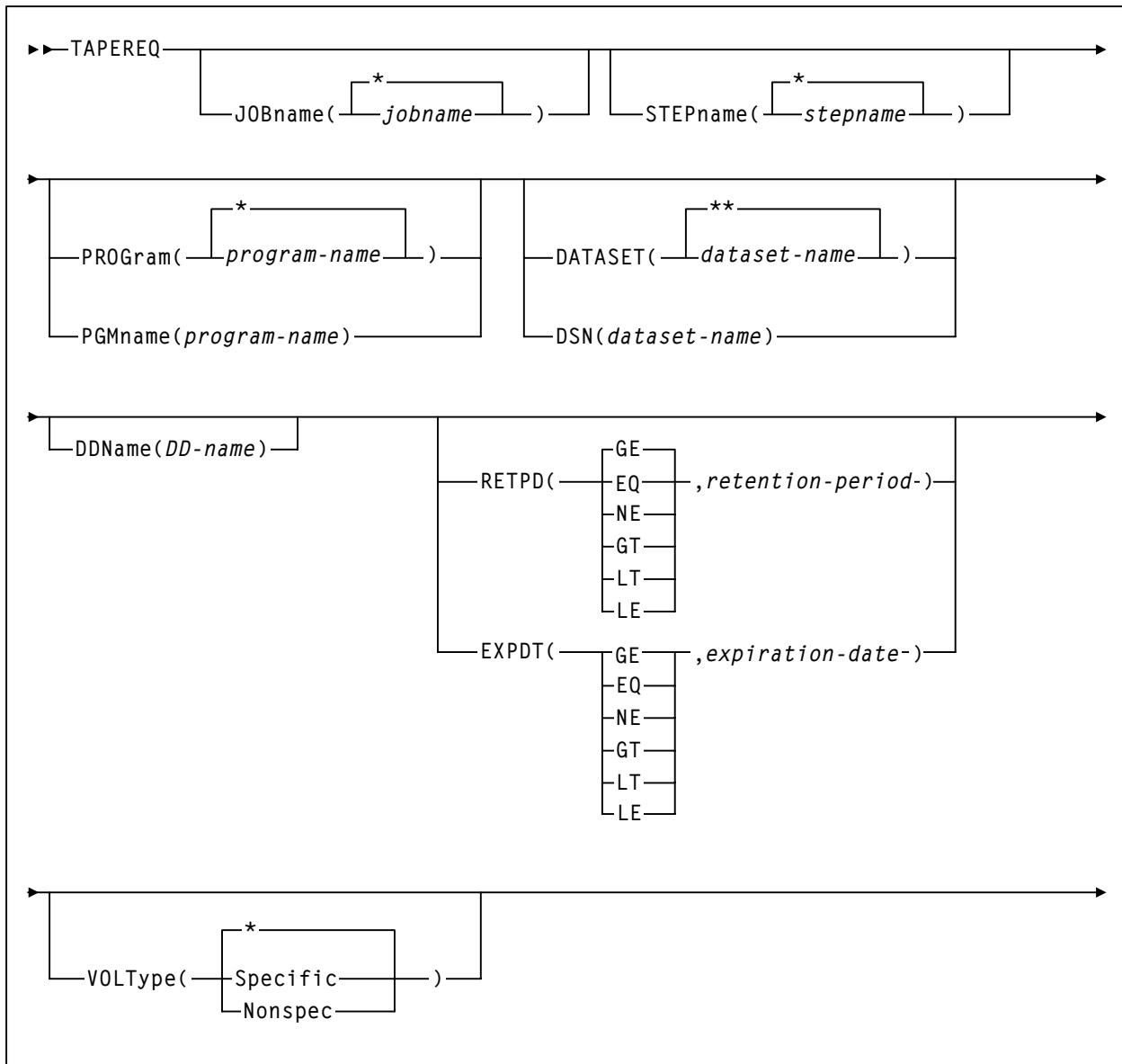
Scratch Subpool Definition (SCRPDEF) command and control statement



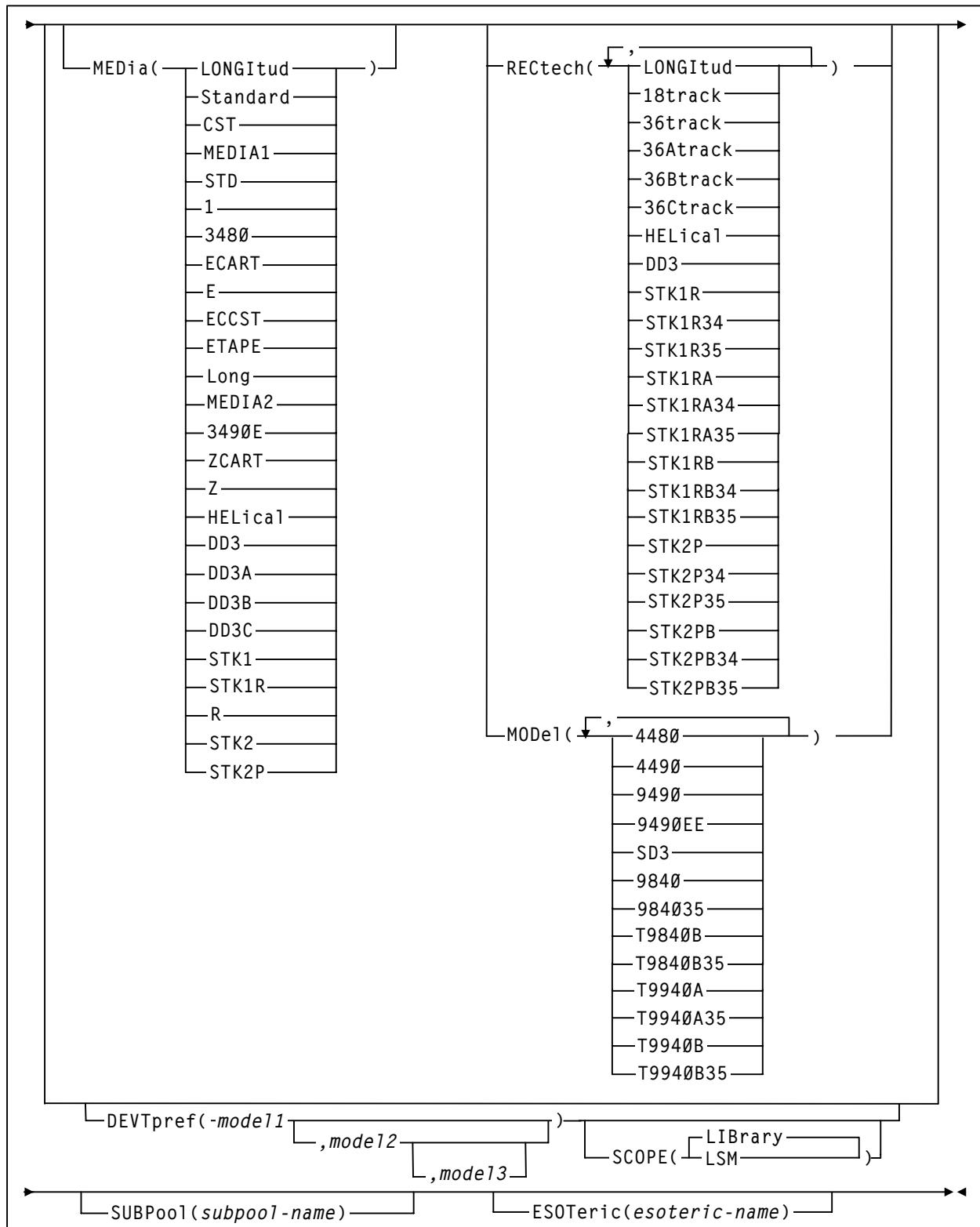
Scratch Subpool (SCRPOol) control statement



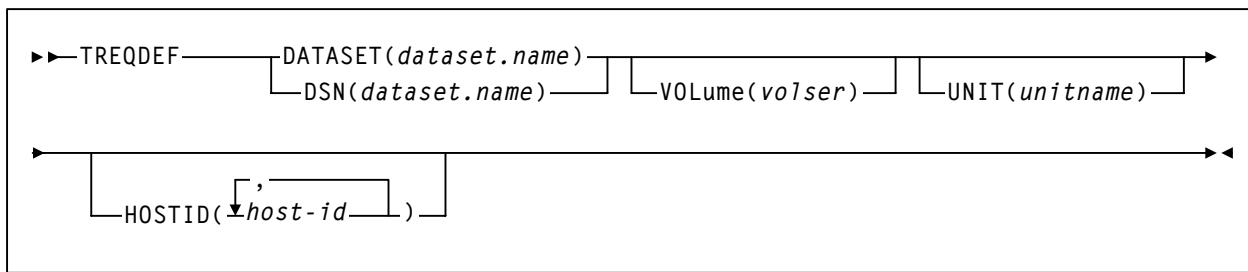
Tape Request (TAPEREQ) control statement



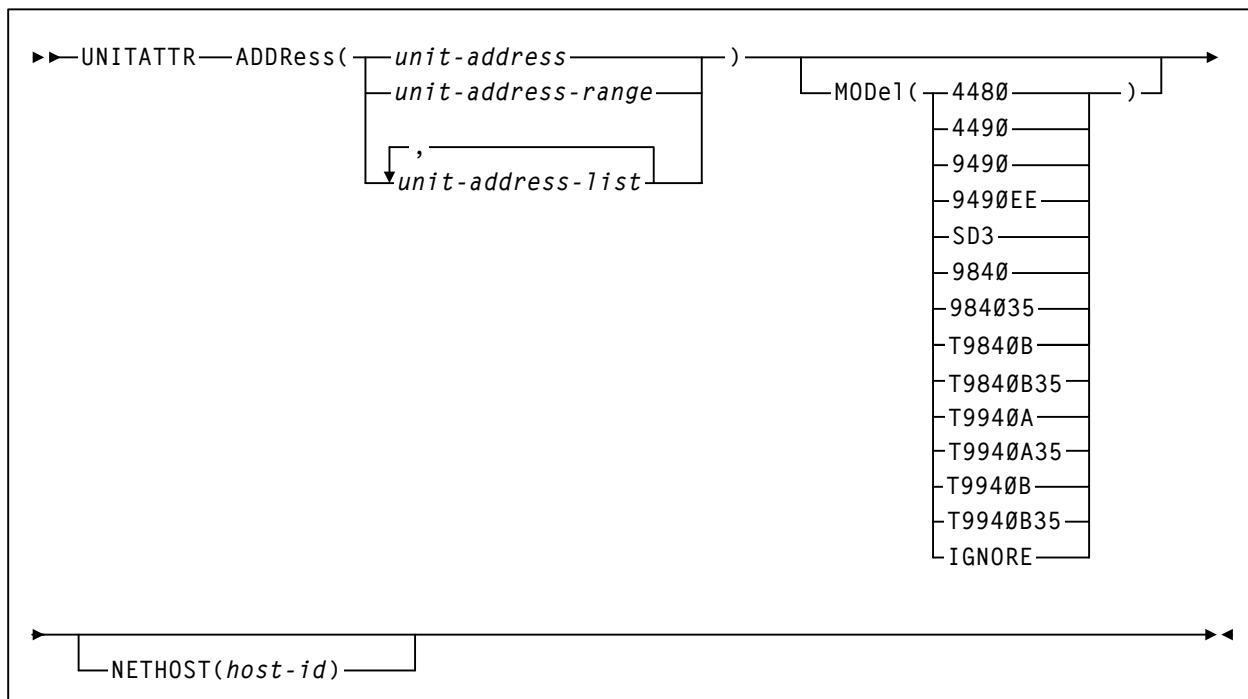
Tape Request (TAPEREQ) control statement (continued)



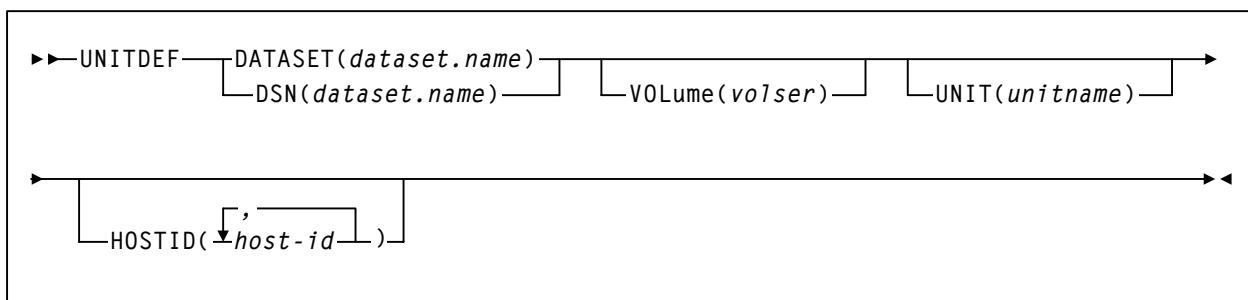
Tape Request Definition (TREQDEF) command and control statement



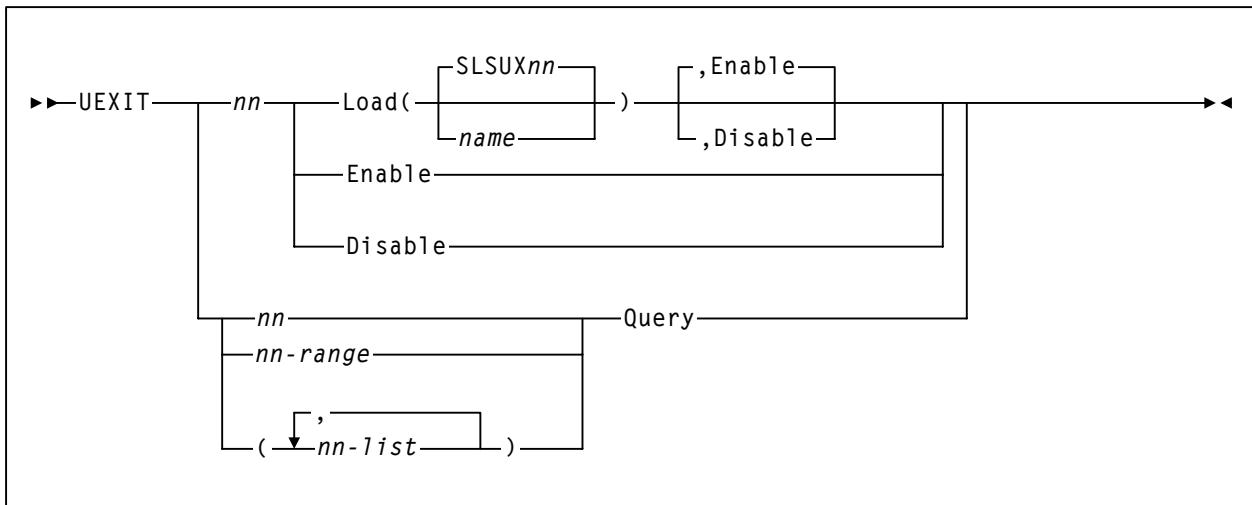
Unit Attribute (UNITATTR) control statement



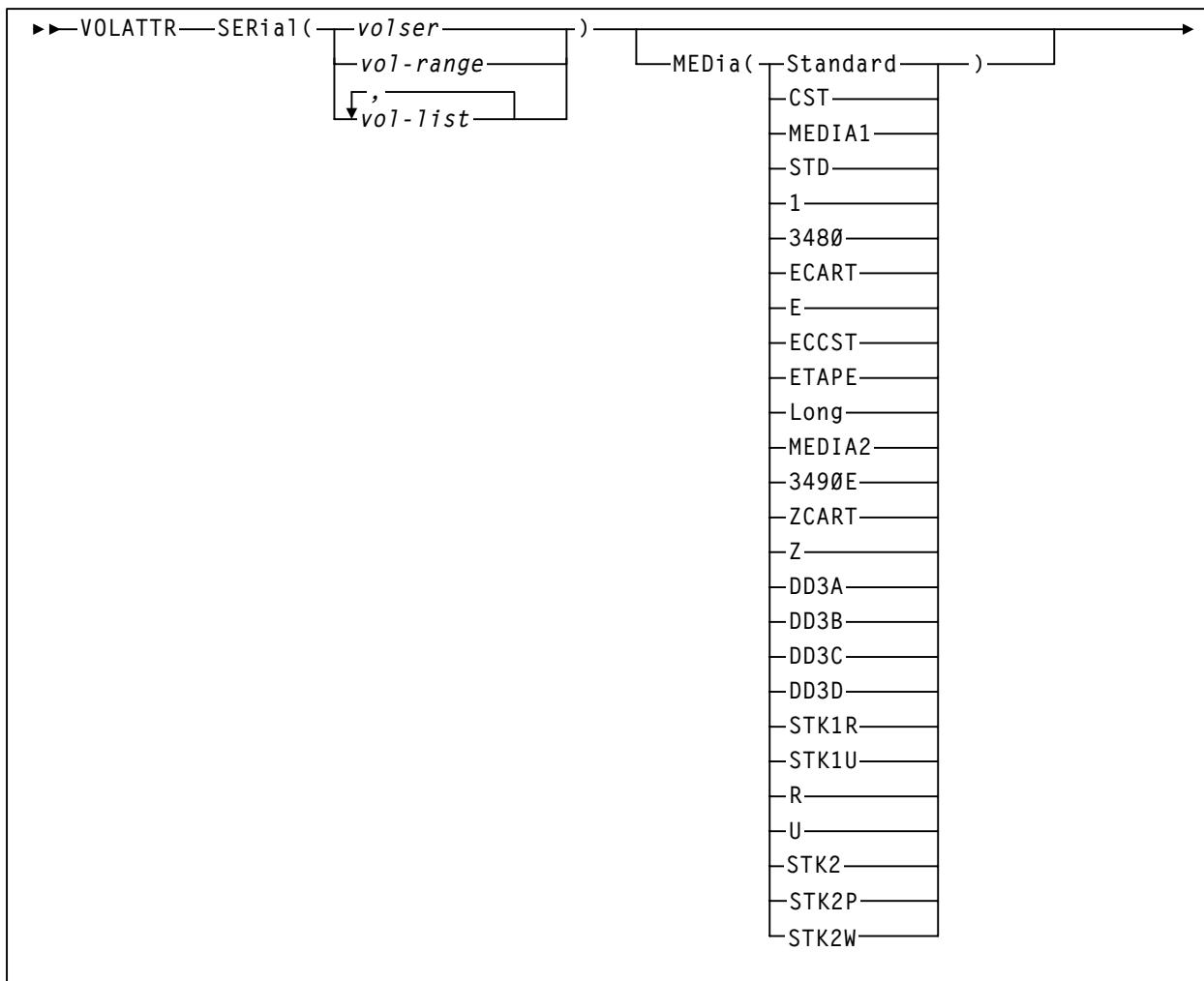
Unit Attribute Definition (UNITDEF) command and control statement



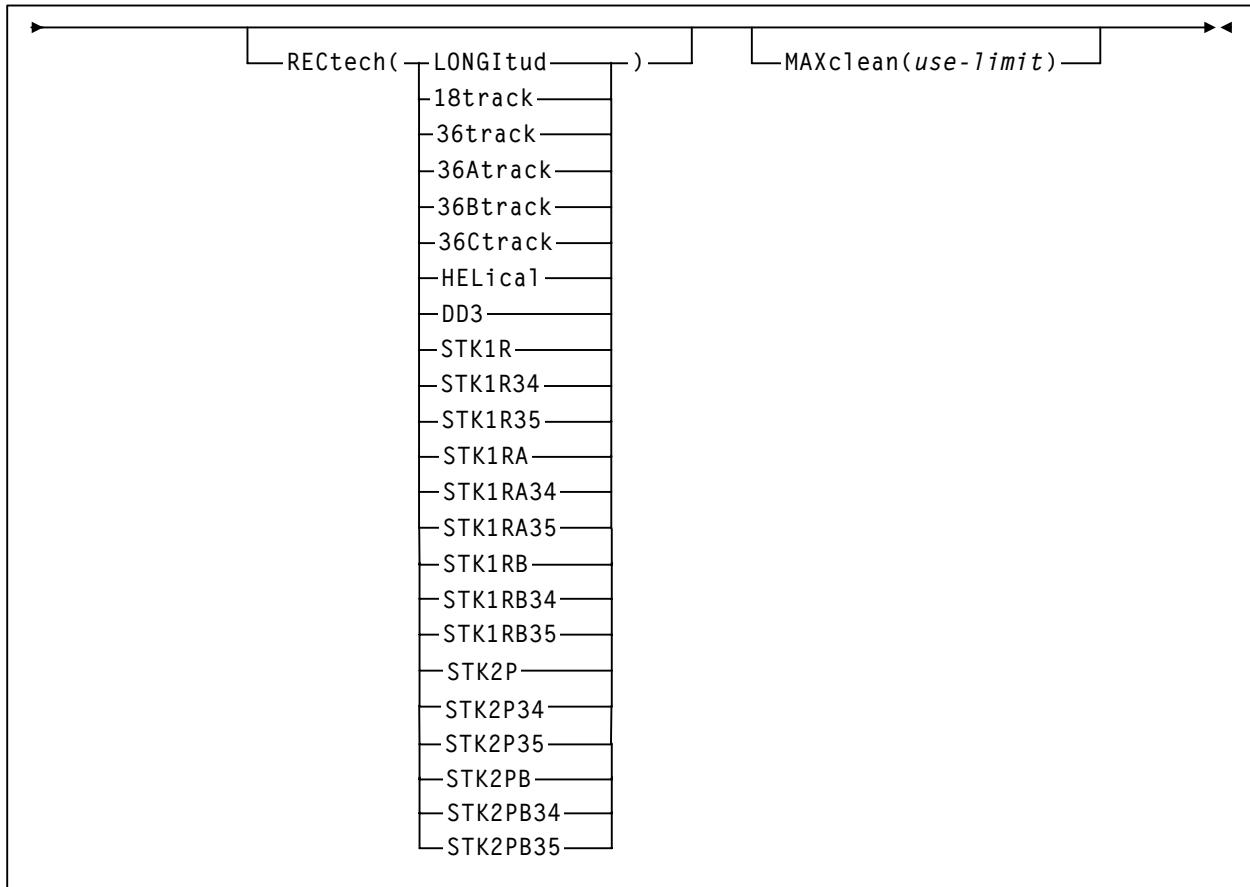
User Exit command and control statement



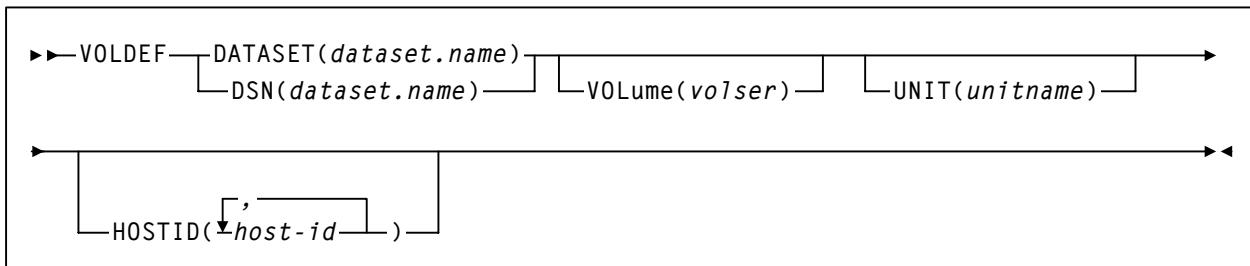
Volume Attribute (VOLATTR) control statement



Volume Attribute (VOLATTR) control statement (continued)



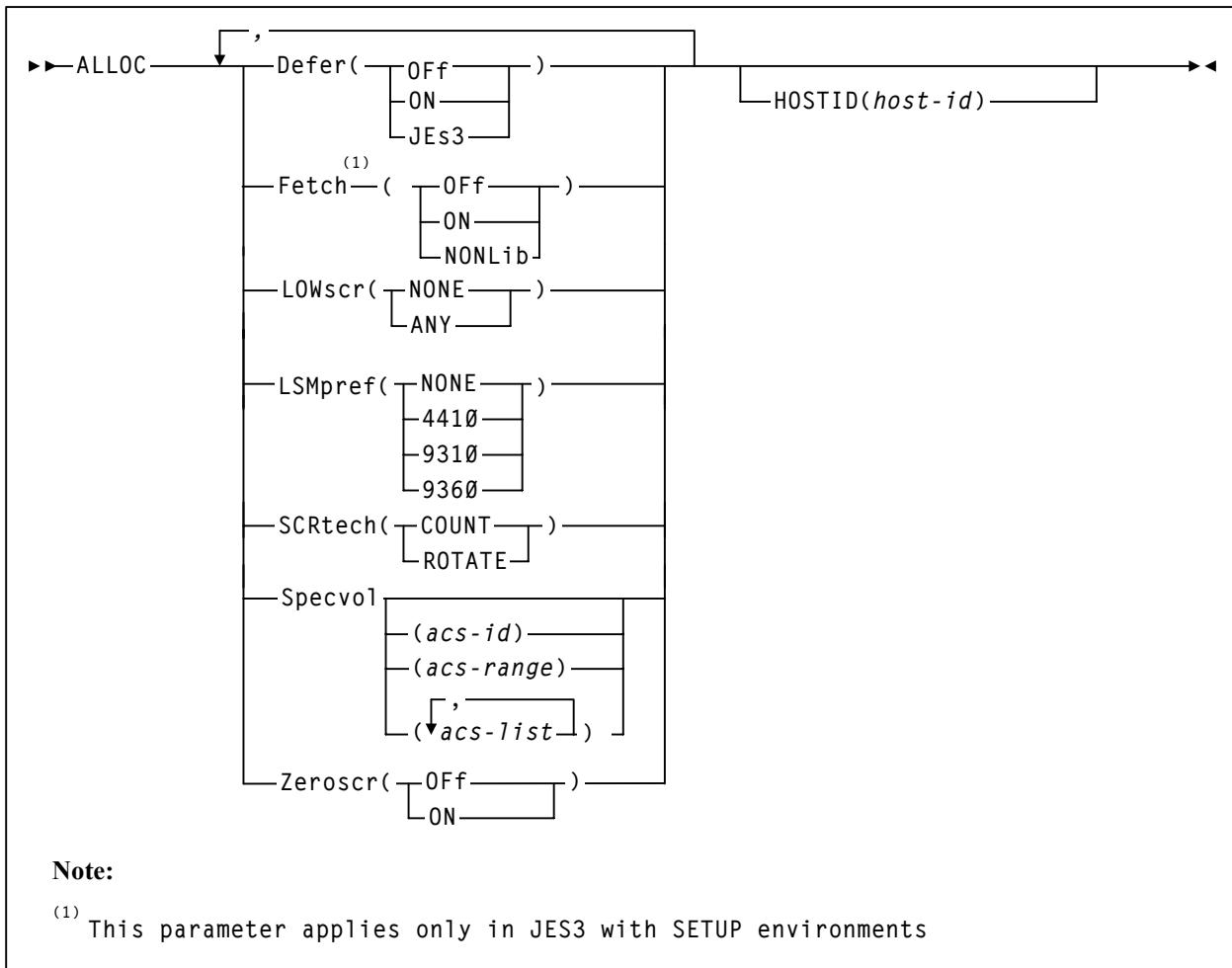
Volume Attribute Definition (VOLDEF) command and control statement



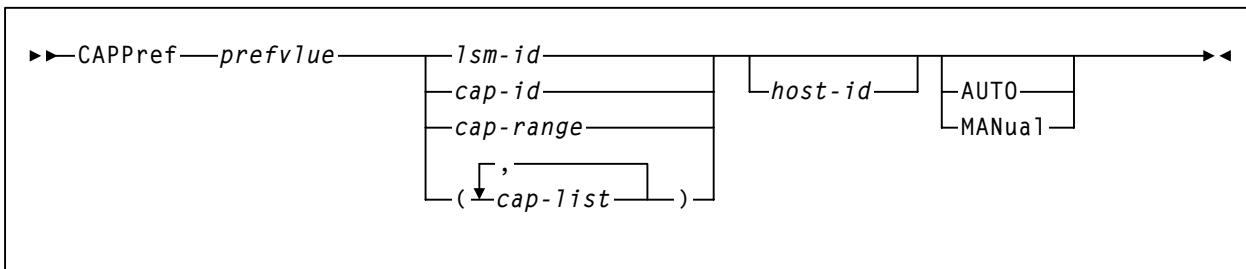
HSC Operator Command Syntax

This section contains the syntax for operator commands. For complete descriptions of the commands, see the *HSC Operator's Guide*.

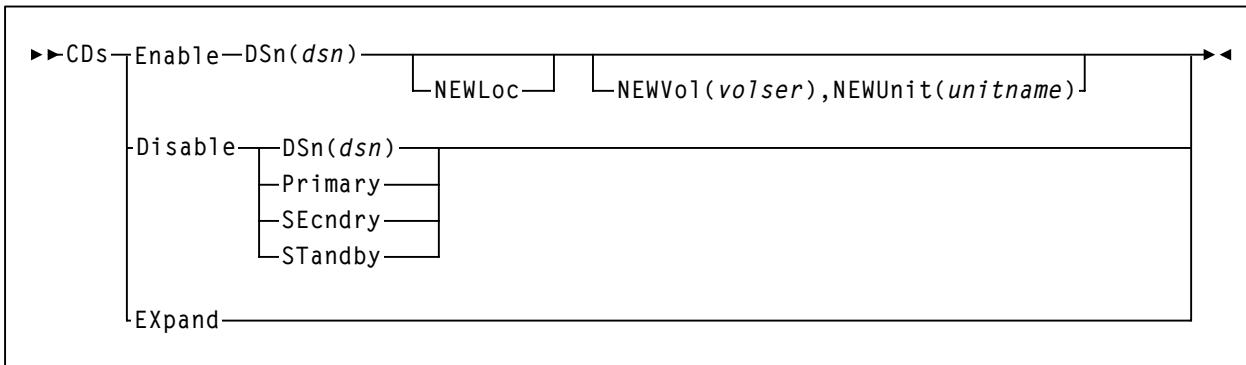
Allocation (ALLOC) command and control statement



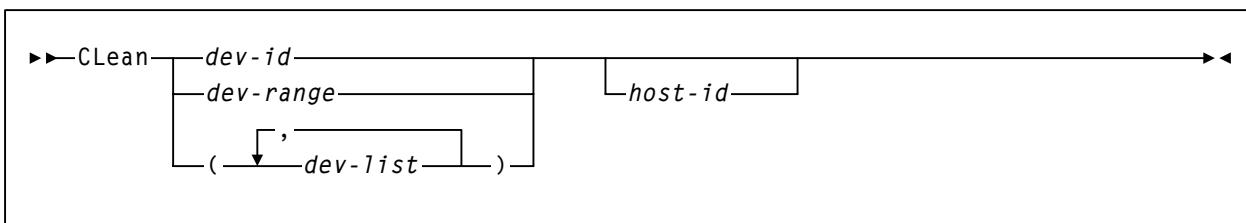
CAP Preference (CAPPref) command and control statement



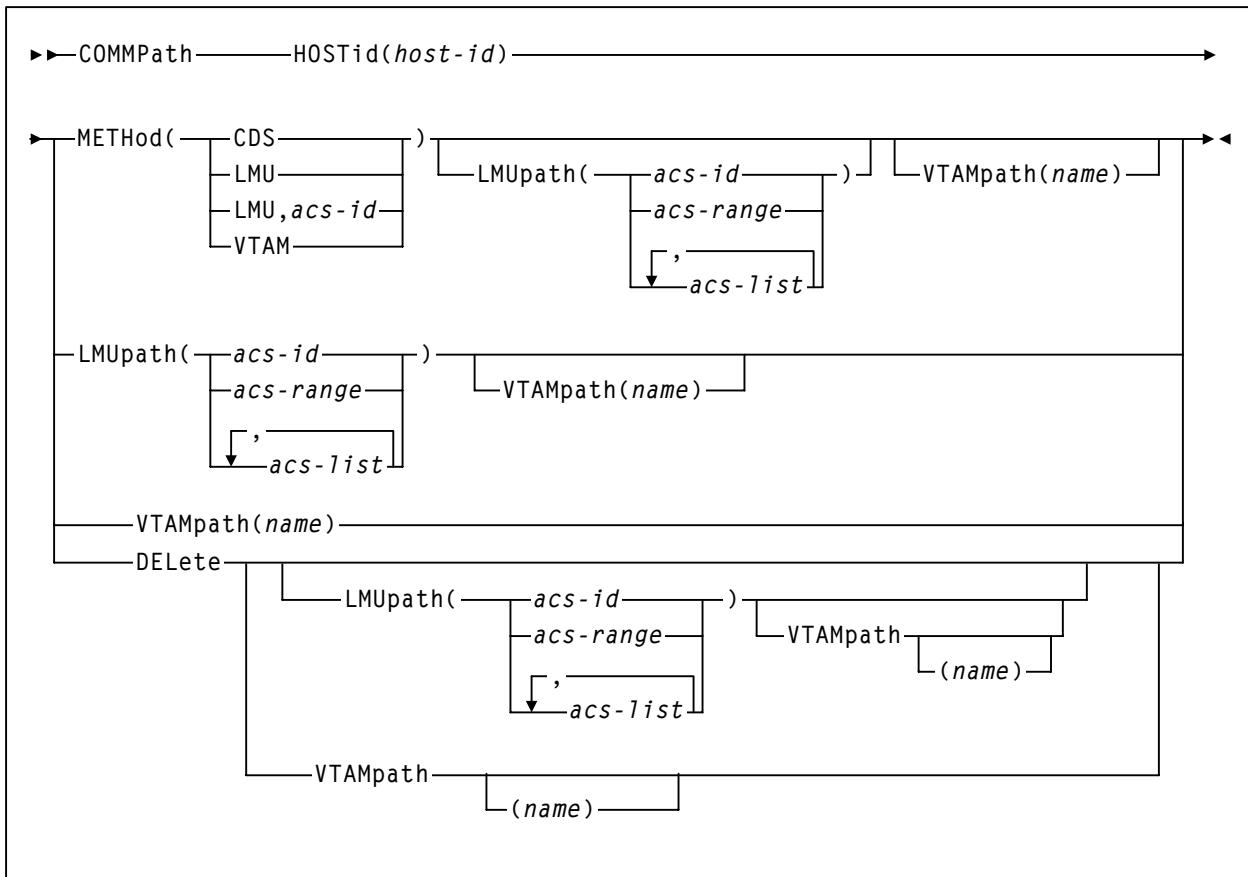
CDs Enable/Disable command



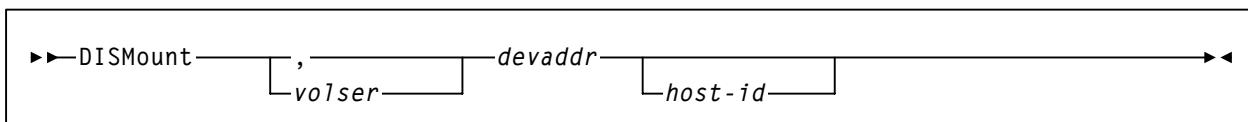
CLean command



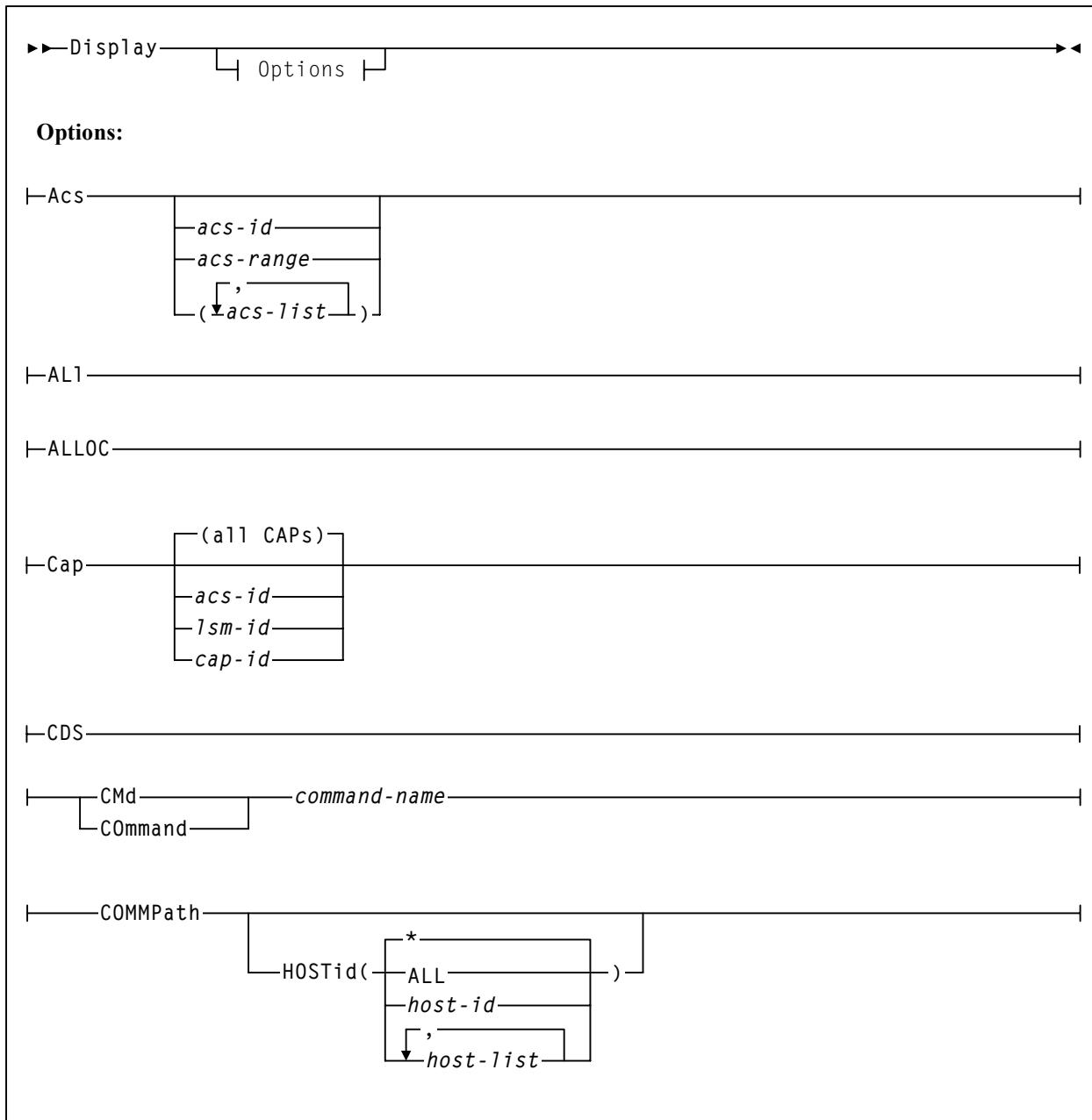
Communications Path (COMMPath) command and control statement



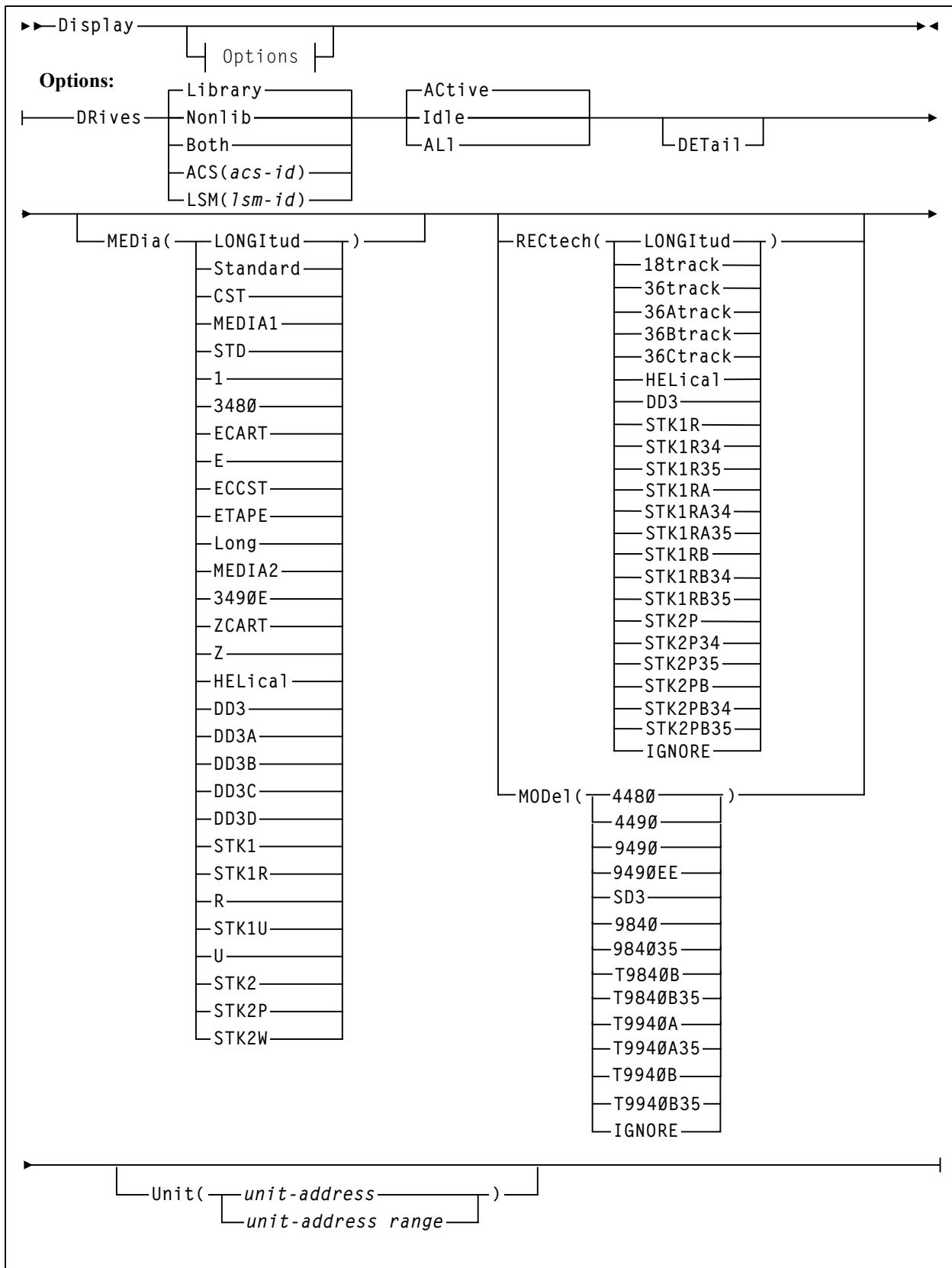
DISMount command



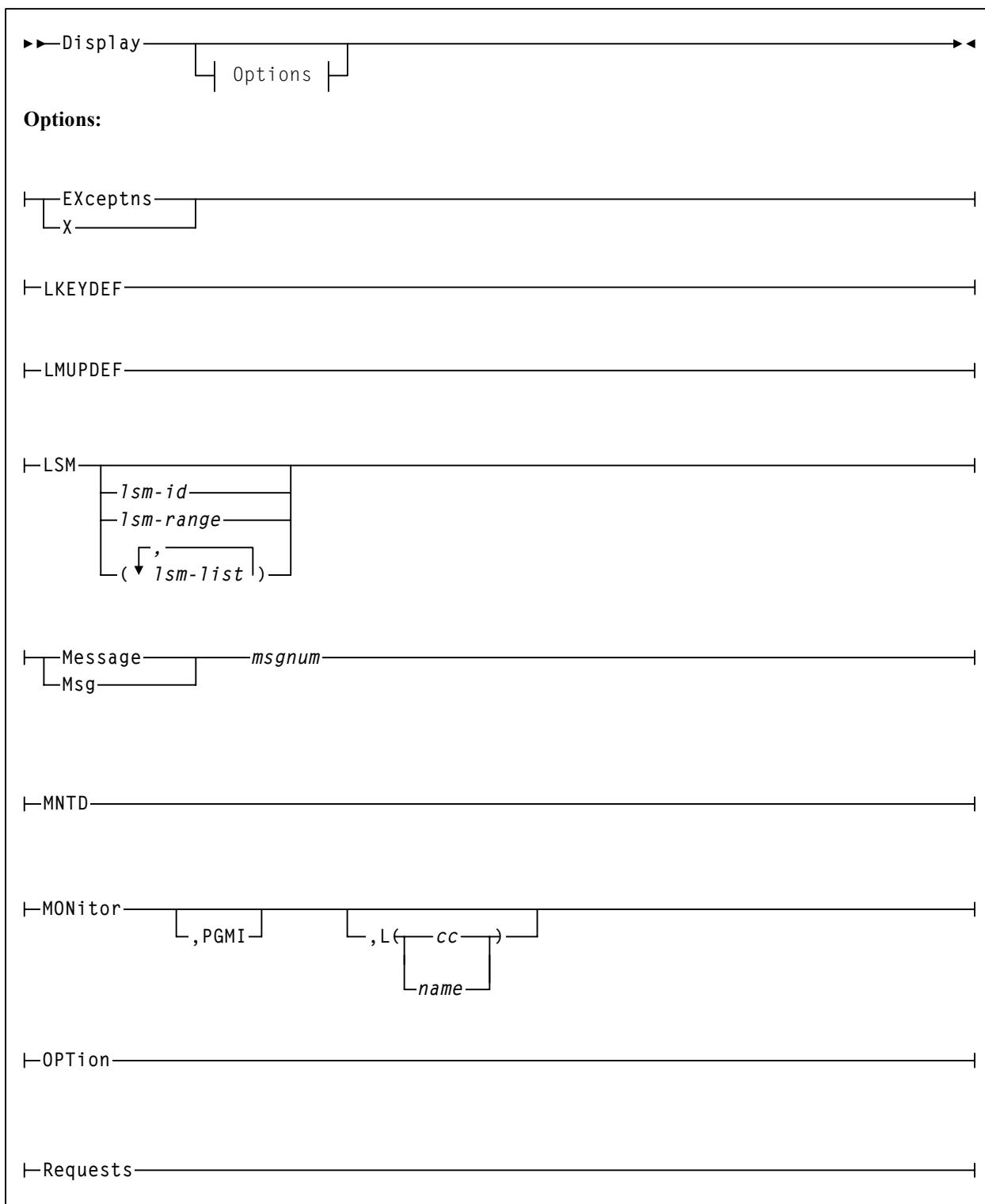
Display command



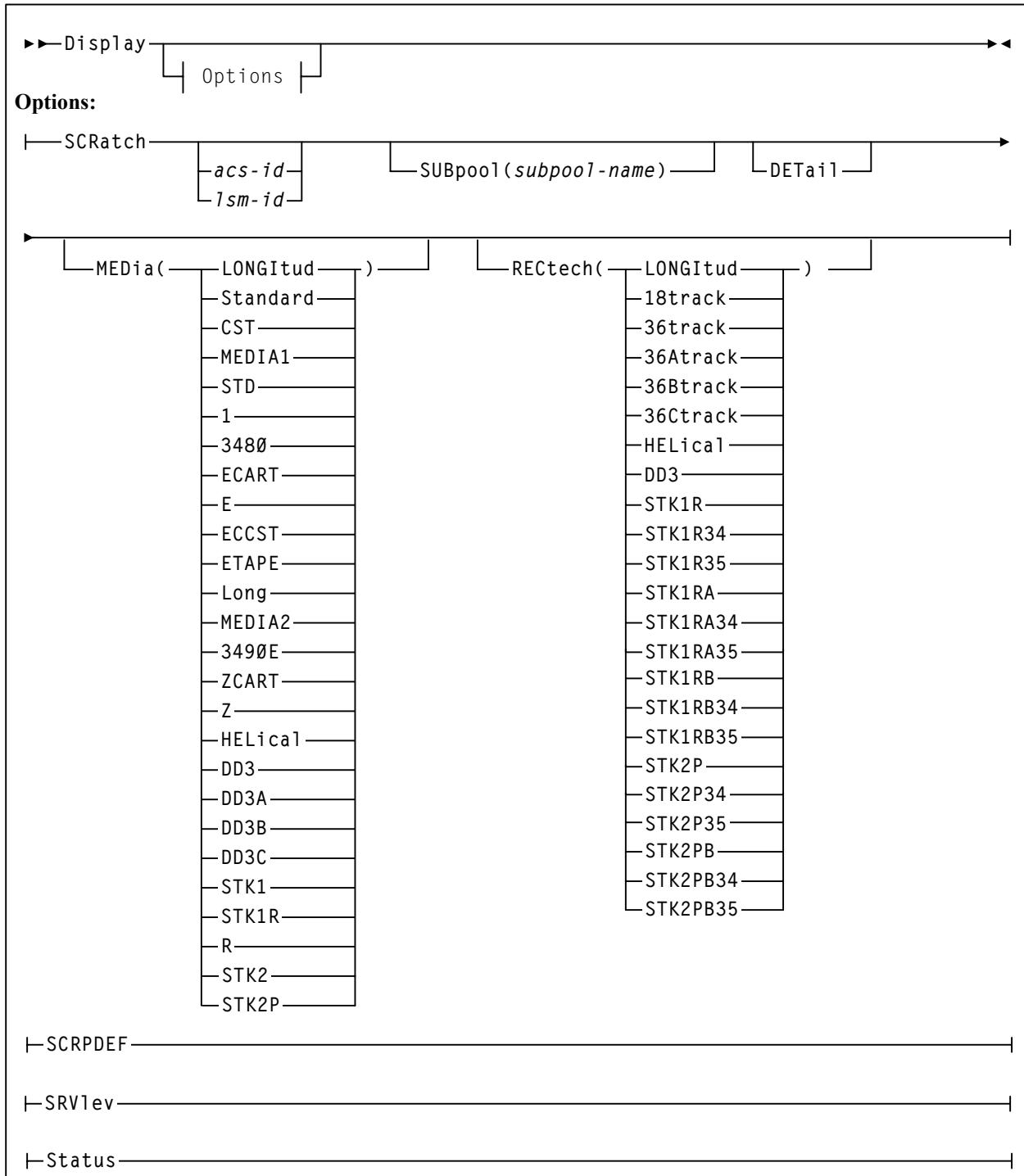
Display command (continued)



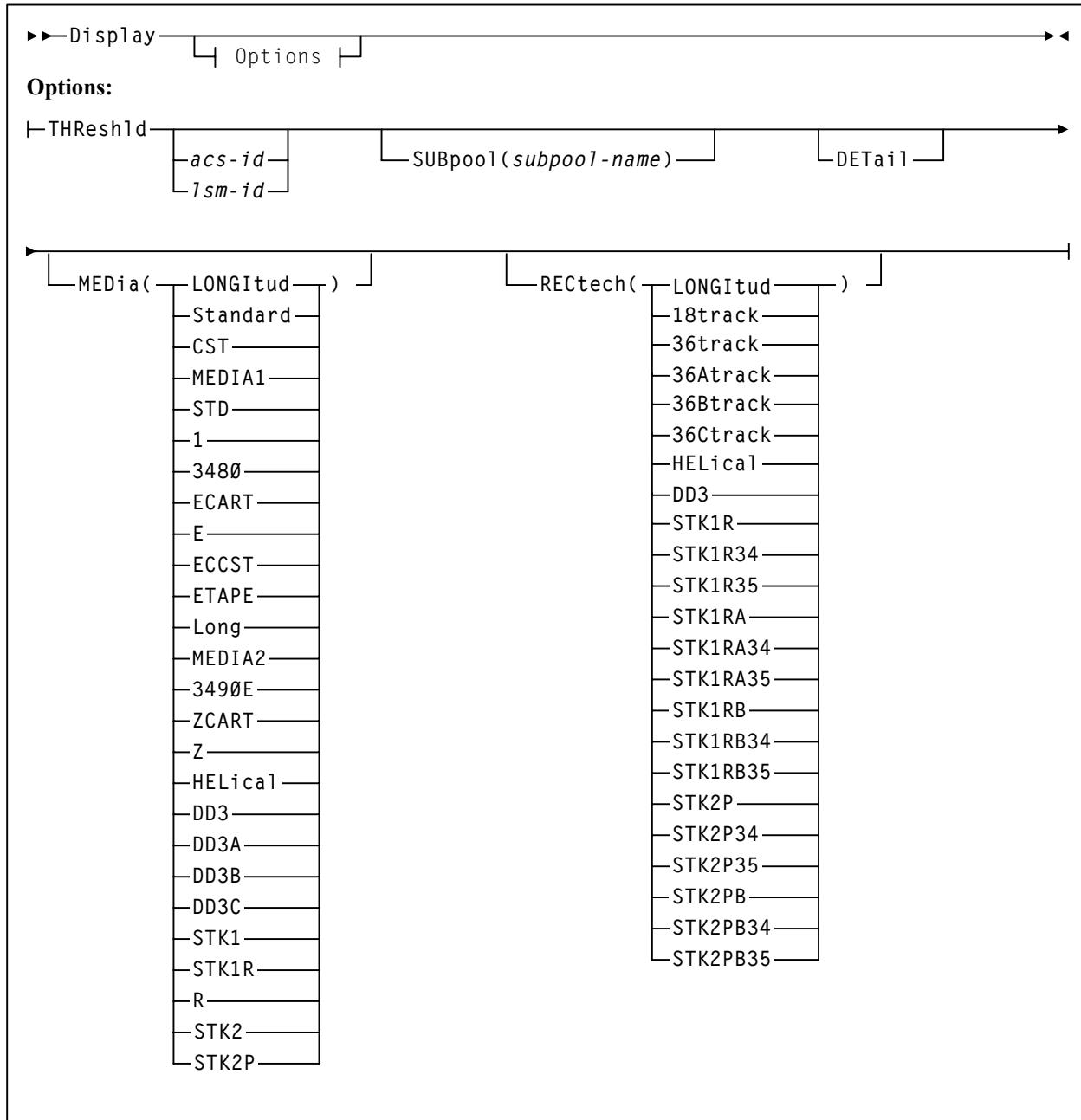
Display command (continued)



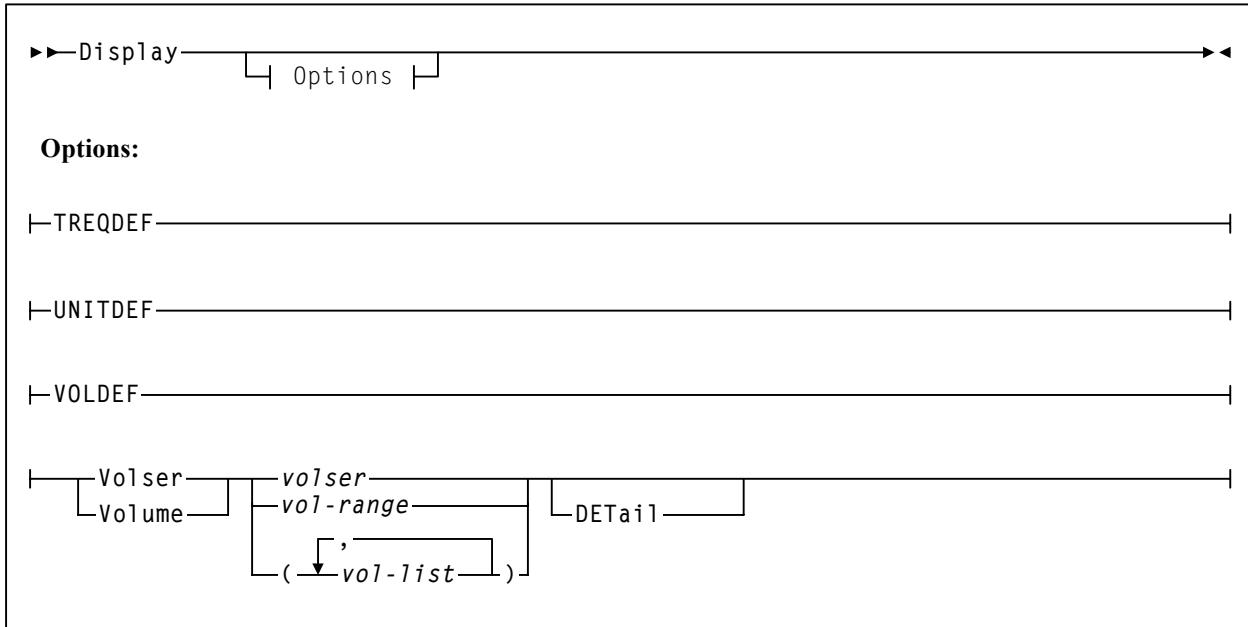
Display command (continued)



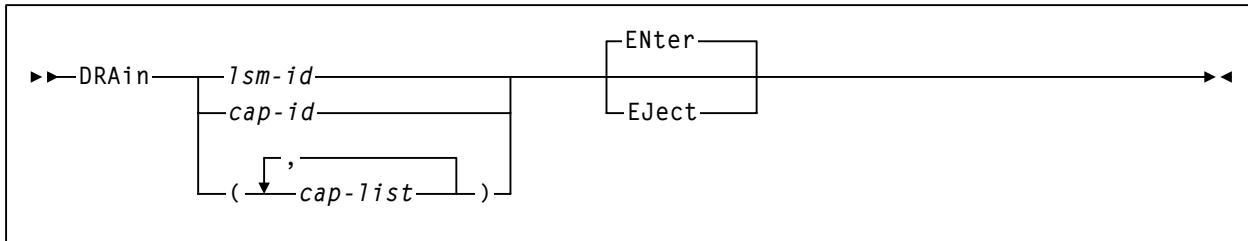
Display command (continued)



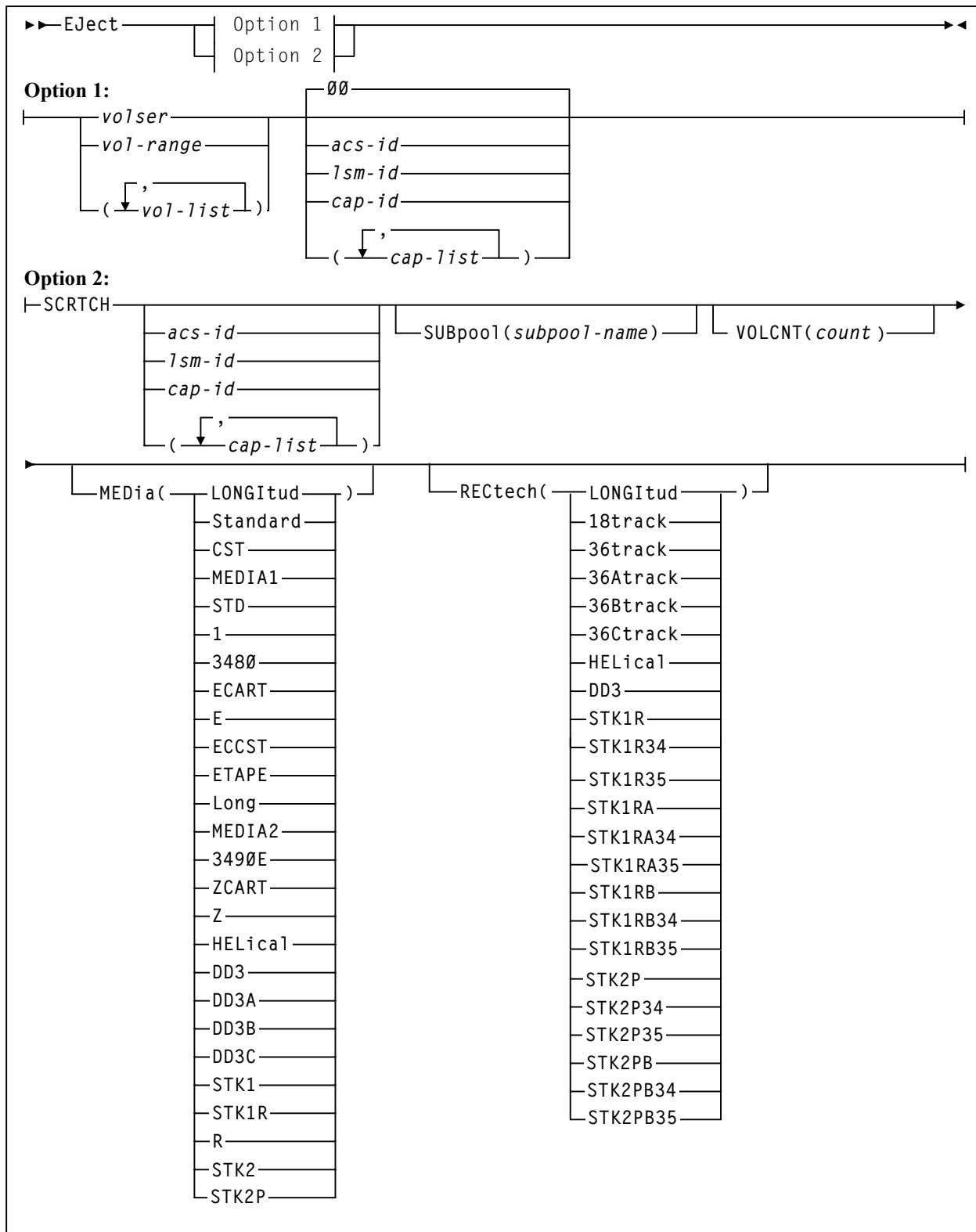
Display command (continued)



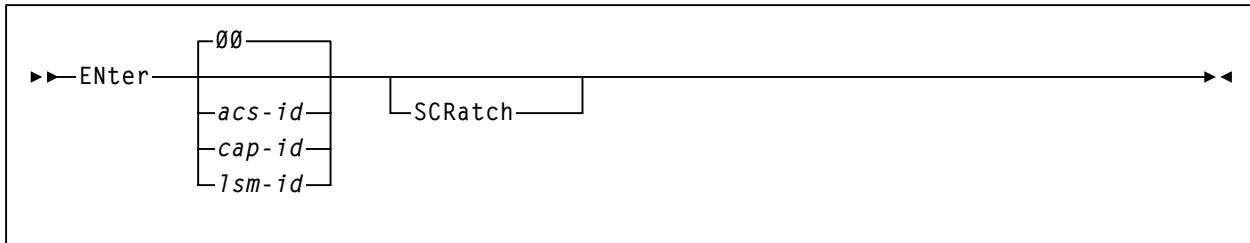
DRAin CAP command



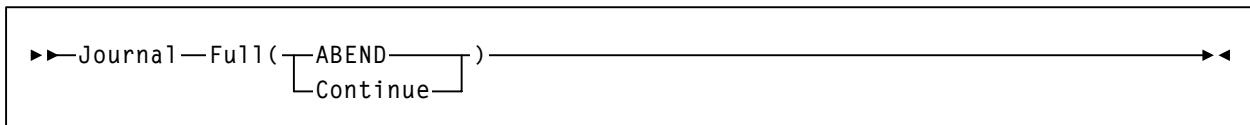
Eject command



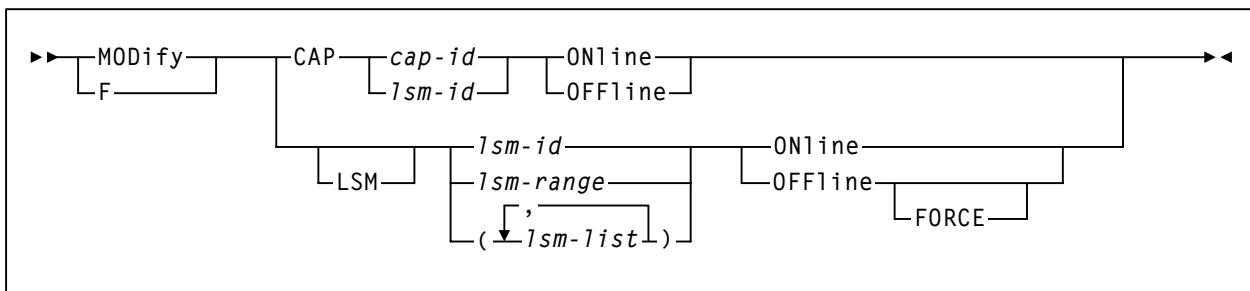
ENter command



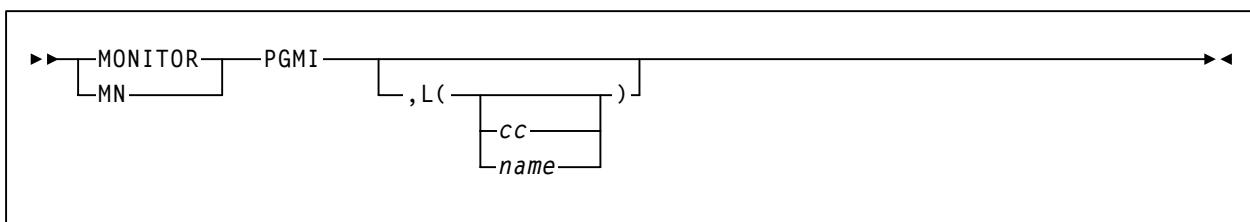
Journal command



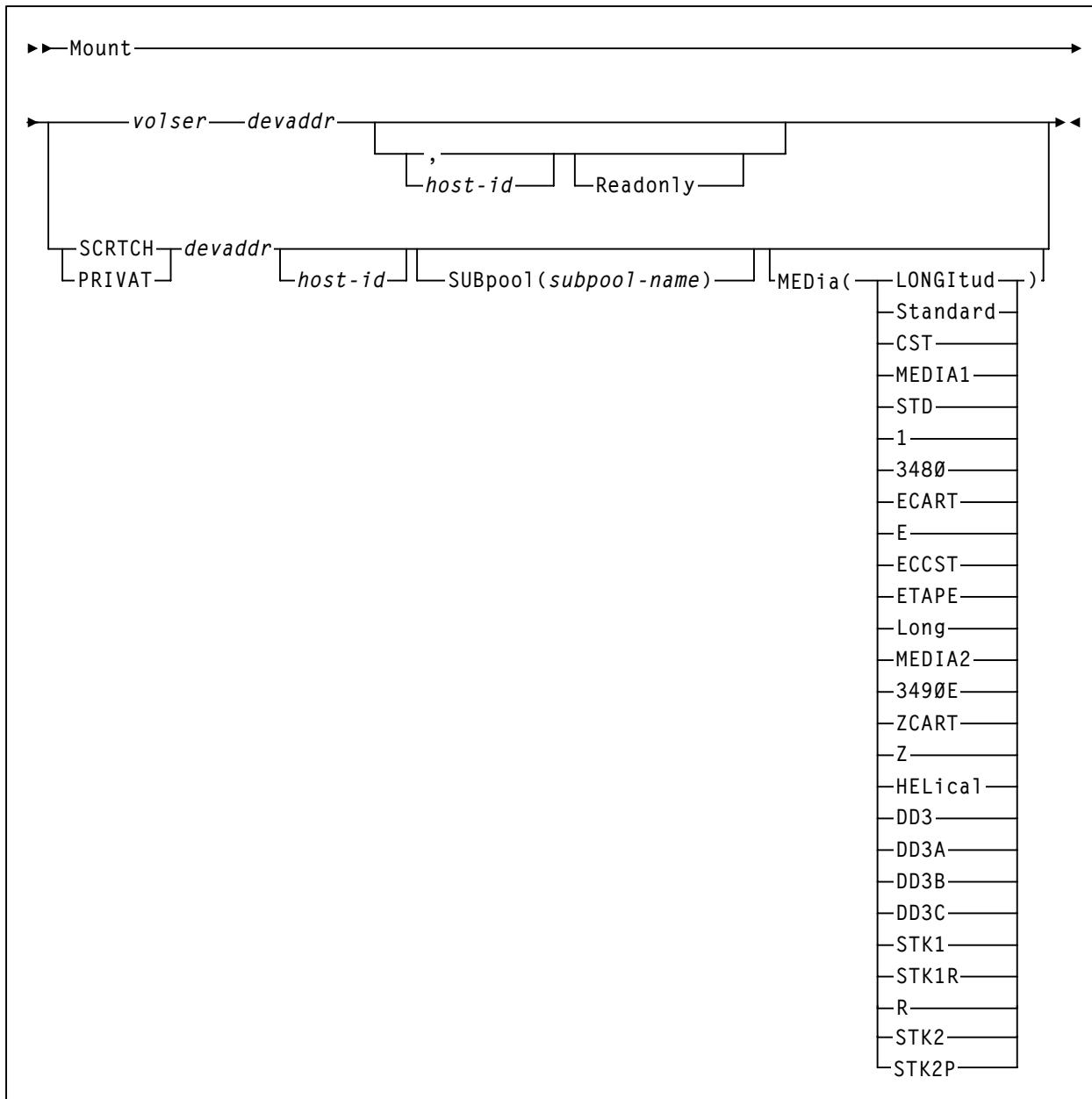
MODify command



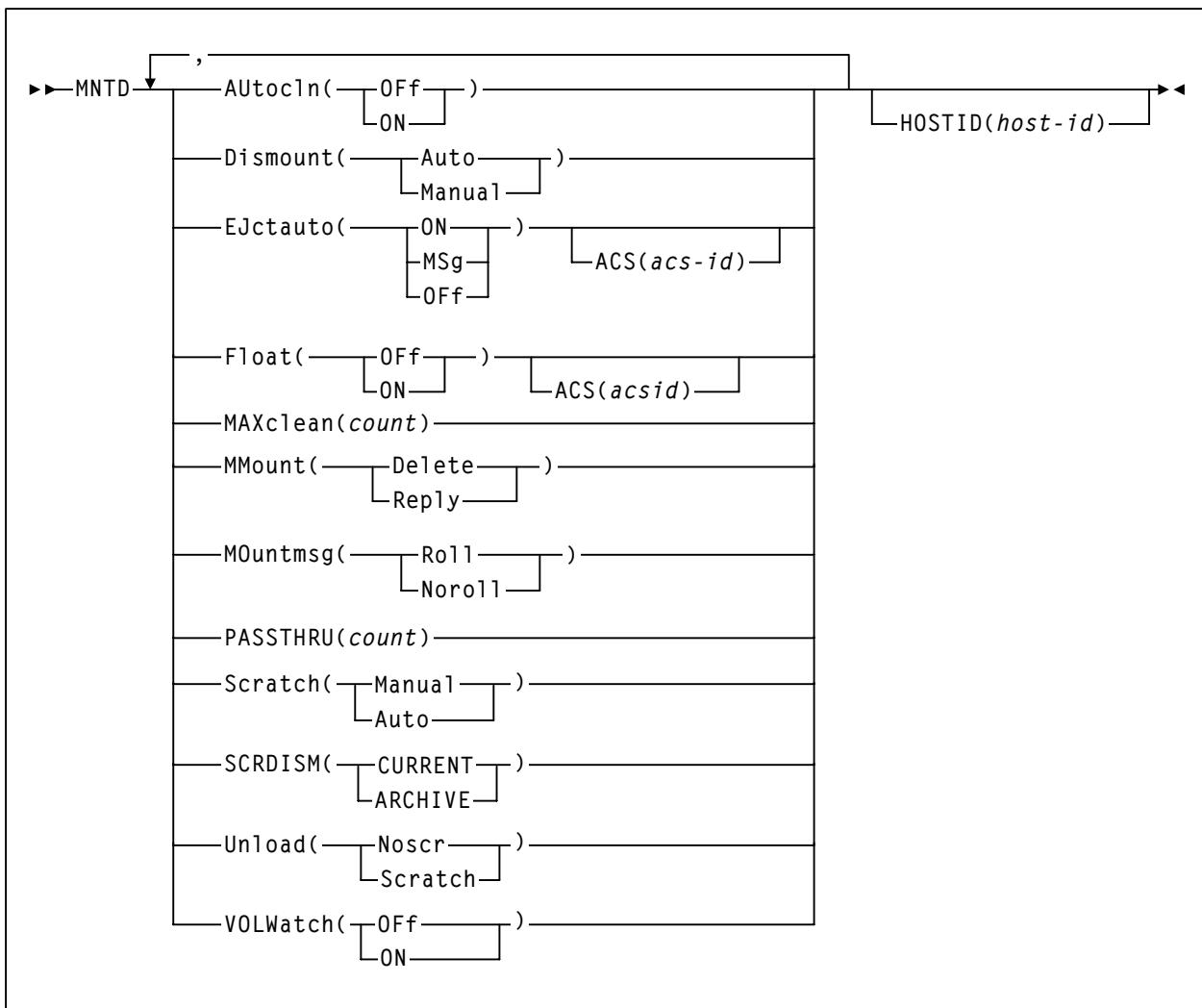
MONITOR command



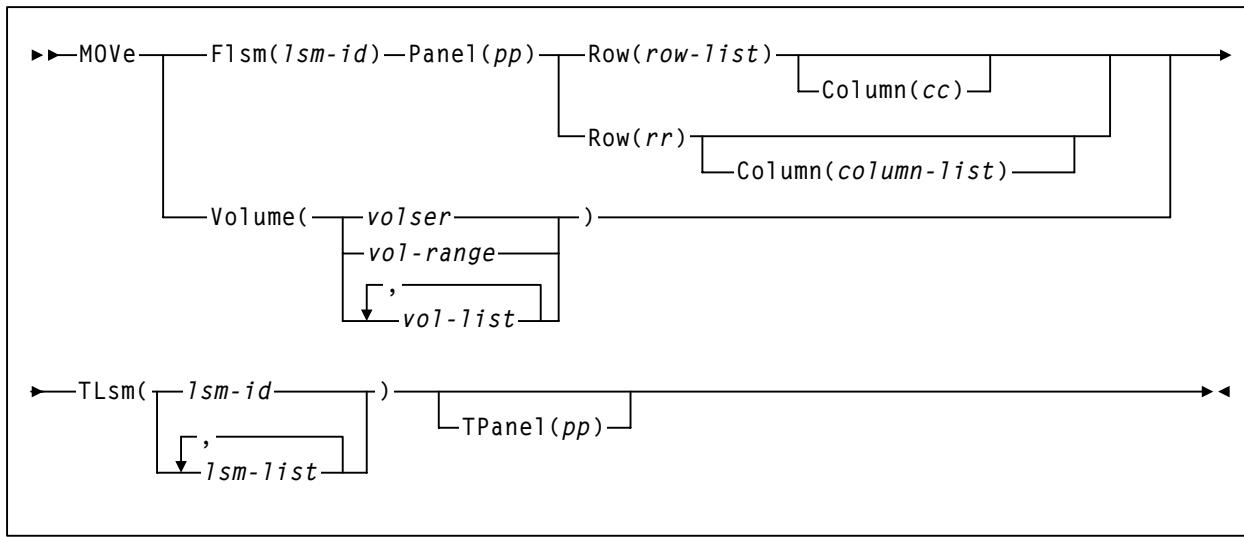
Mount command



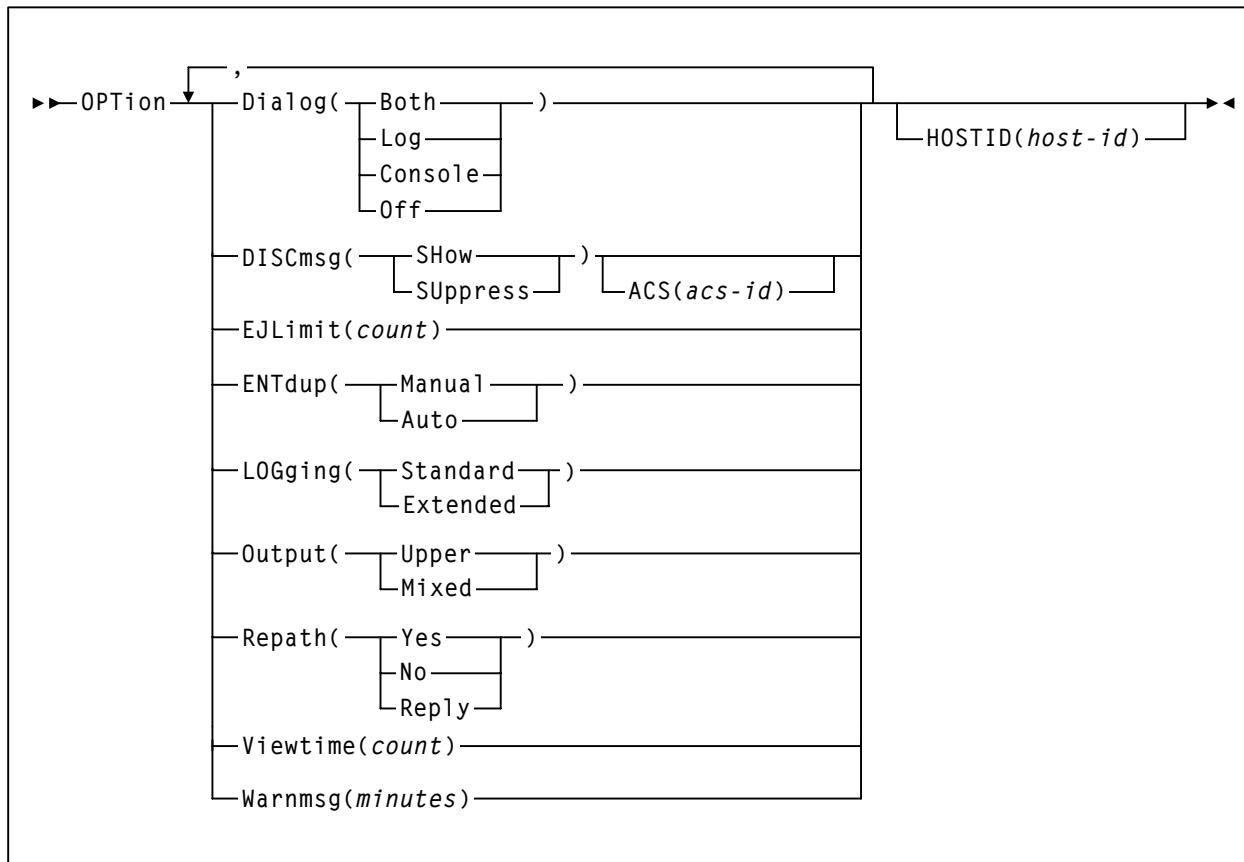
Mount/Dismount Options (MNTD) command and control statement



MOVE command



OPTION command and control statement



RECover Host command

```
►►RECover—host-id—  
          └FORCE┘
```

RELEASE CAP command

```
►►RELEASE—cap-id—
```

SENter command

```
►►SENter—cap-id—  
          └lsm-id┘
```

SRVlev command

```
►►SRVlev—BASE—  
          └FULL┘
```

Stop Monitoring (STOPMN) command

```
►►STOPMN—PGMI—  
      └PM┘—  
          └, L(—  
            └cc—  
            └name┘—  
          ┘—
```

SWitch command

```
►►SWitch [Acs acs-id(1)]
```

Note:

(1) ACS *acs-id* is optional in a single-ACS environment; it is required in a multiple-ACS environment.

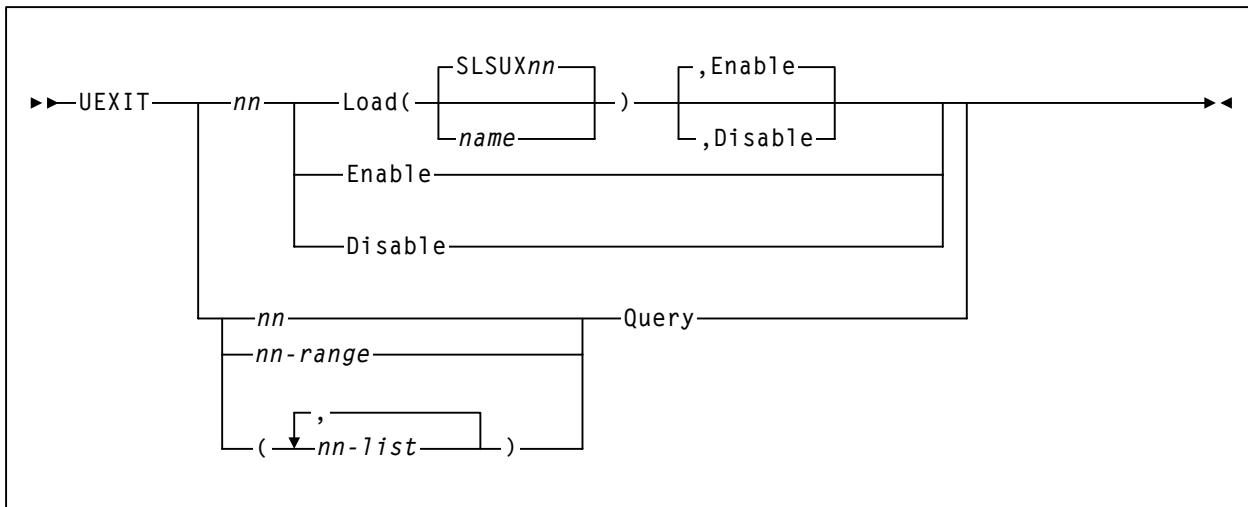
TRace command

```
►►TRace [comp-name]  
        [, comp-list]  
OFF [comp-name]  
      [, comp-list]
```

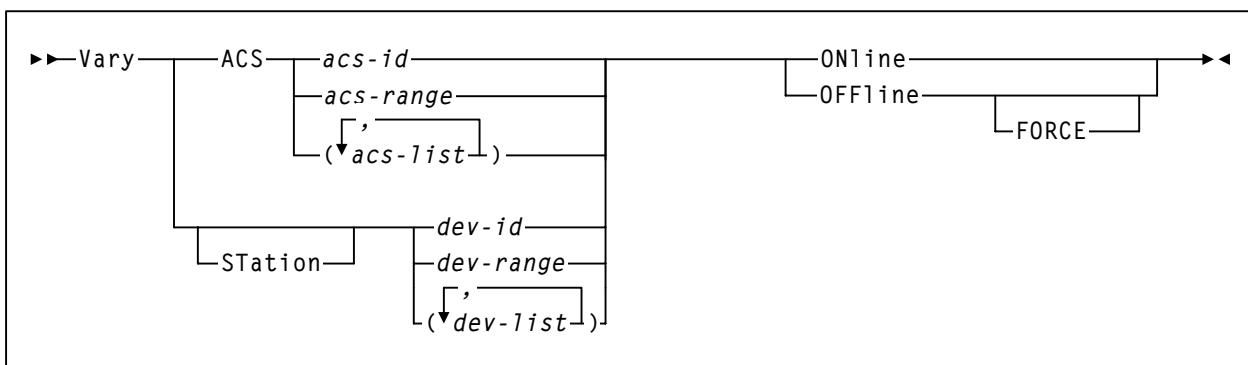
TRACELKP command

```
►►TRACELKP [table-name]  
         [, table-list]  
OFF [table-name]
```

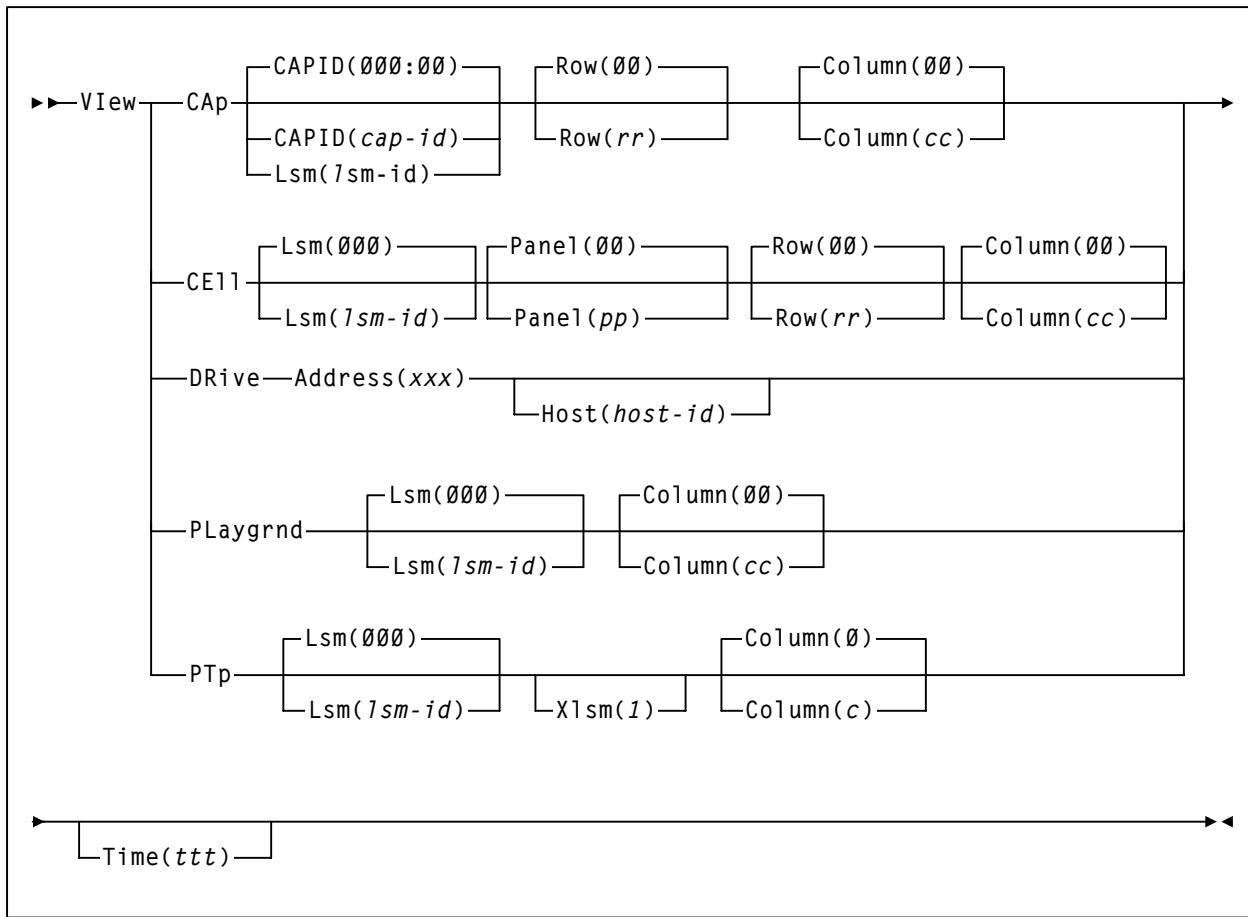
User Exit (UEXIT) command and control statement



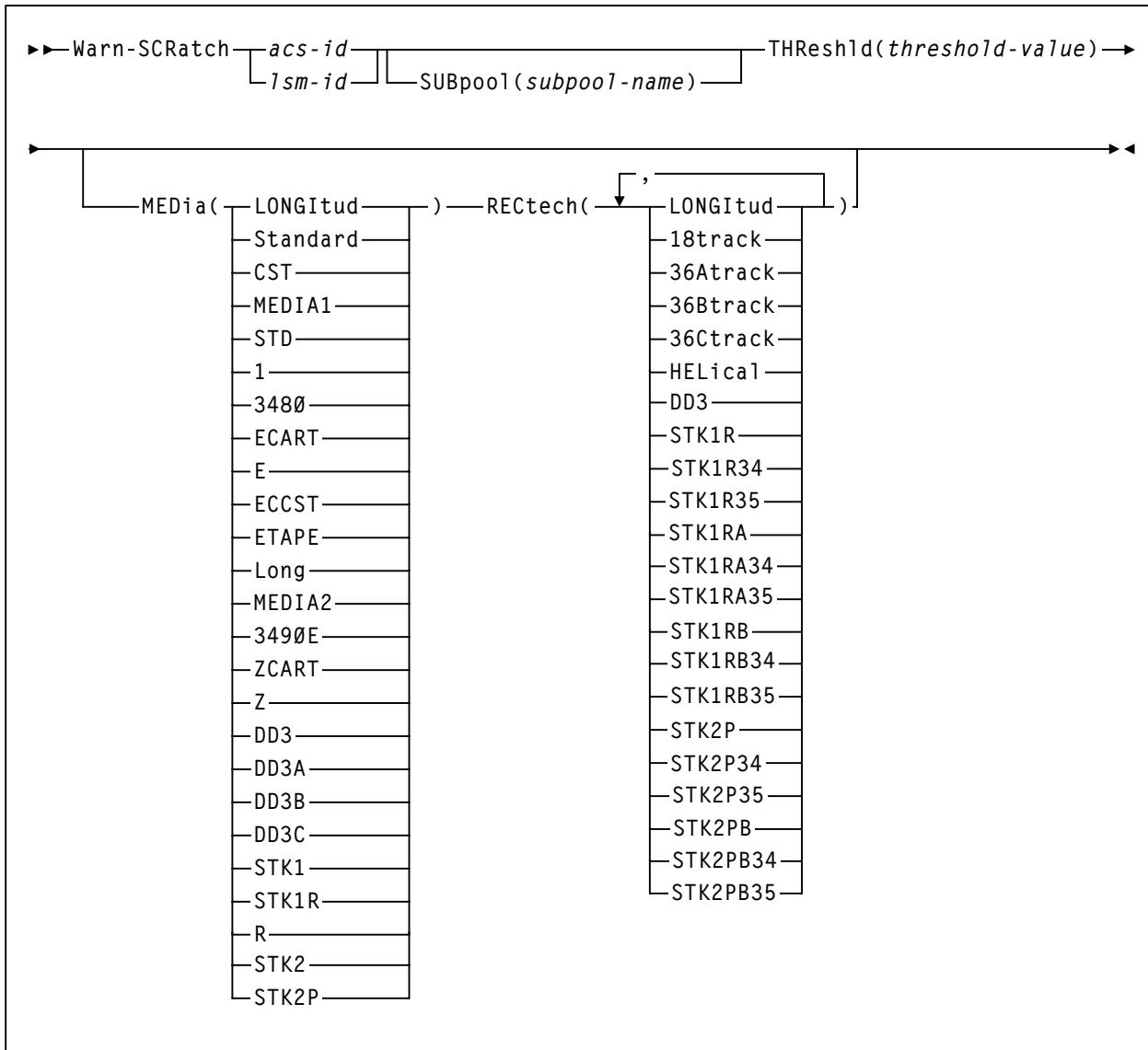
Vary Station command



View command



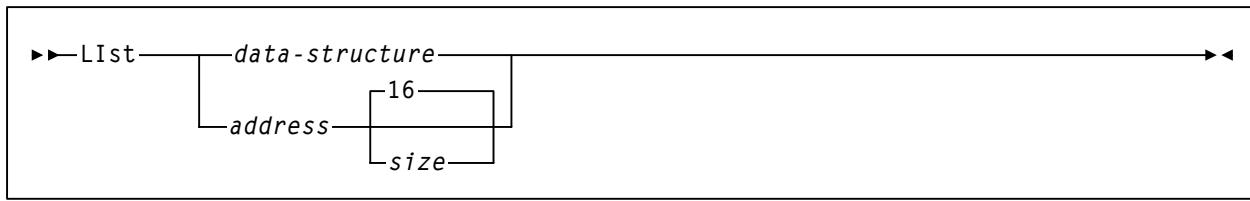
Warn command



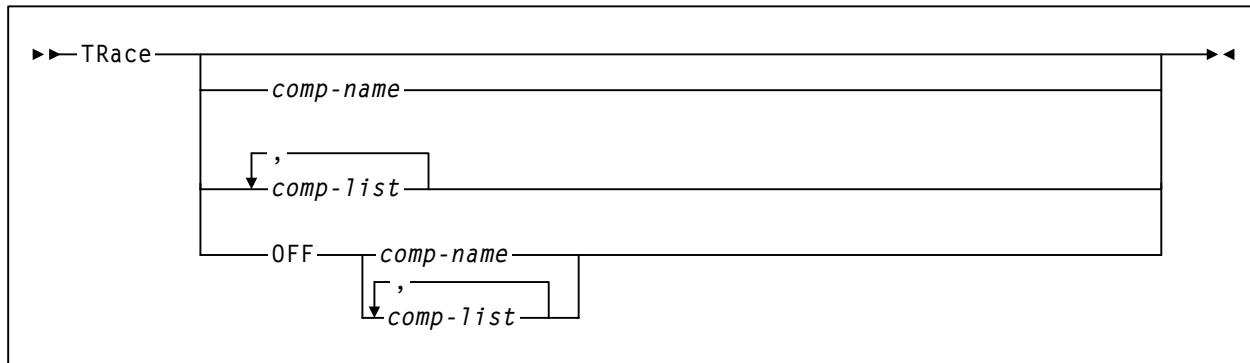
HSC Diagnostic Command Syntax

This section contains the syntax for diagnostic commands. For complete descriptions of the commands, see the *HSC System Programmer's Guide* and *HSC Operator's Guide*.

List command



TRace command



Printed in U.S.A.