

StorageTek Host Software Component (HSC) MSP Implementation

Reference Summary



August 2010
Revision BA

Submit comments about this document by clicking the Feedback [+] link at: <http://docs.sun.com>

Host Software Component (HSC) Reference Summary

Copyright ©1987, 2010, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related software documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.



Document Effectivity

EC Number	Date	Doc Kit Number	Type	Effectivity
132246	May, 2006	---	Revision A	This document applies to the Host Software Component for MSP, version 6.1.
---	February, 2009	---	Revision B	
---	August, 2010	---	Revision BA	

Contents

Preface	ix
Related Documentation	x
Documentation, Support, and Training	xi
Oracle Welcomes Your Comments	xi
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	7
Ranges and Lists	9
Control Statement Syntax Conventions	12
LIBGEN Macro Syntax	15
SLIACS macro	15
SLIALIST macro	15
SLIDLIST macro	15
SLIDRIVS macro	16
SLIENDGN macro	16
SLILIBRY macro	16
SLILSM macro	17
SLIRCVRY macro	17
SLISTATN macro	18
Control Statement Syntax	19
Control Data Set Definition (CDSDEF) control statement	19
EXECParm control statement	19
Journal Definition (JRNDEF) control statement	20

LMUPATH control statement	20
LMUPDEF command and control statement	20
OPTion control statement	20
Reconfiguration Definition (RECDEF) control statement	21
Scratch Subpool Definition (SCRPDEF) command and control statement	21
Scratch Subpool (SCRPOol) control statement	21
Volume Attribute (VOLATTR) control statement	22
Volume Attribute Definition (VOLDEF) command and control statement	22
Utility Syntax	23
ACTivities Report utility	23
AUDIt utility	24
BACKup utility	24
Database Decompile (LIBGEN) utility	24
Directory Rebuild (DIRBLD) utility	24
EJECT utility	25
INITialize Cartridge utility	25
Journal OFFLoad utility	25
MERGEcds utility	26
MOVE utility	26
Reconfiguration utility	26
REPLace utility	26
RESTore utility	27
SCRAtch utility	27
Scratch Conversion utility	27
Scratch Redistribution (SCREdist) utility	28
SET utility	29
UNSCRatch utility	30
UNSElect utility	30
Volume Report (VOLRpt) utility	31
HSC Operator Command Syntax	33
Allocation (ALLOC) command and control statement	33
CAP Preference (CAPPref) command and control statement	33
CDs Enable/Disable command	34
CLean command	34
Communications Path (COMMPath) command and control statement	35
DISMount command	35
Display command	36
DRAin CAP command	41
EJect command	41
ENter command	41
Journal command	42
MODify command	42
MONITOR command	42
Mount command	43

Mount/Dismount Options (MNTD) command and control statement	43
MOVE command	44
OPTION command and control statement	45
RECOVER Host command	45
RELEASE CAP command	45
SENTER command	46
SRVlev command	46
Stop Monitoring (STOPMN) command	46
SWITCH command	46
TRACE command	47
TRACELKP command	47
User Exit (UEXIT) command and control statement	47
Vary Station command	48
VIEW command	48
WARN command	49
HSC Diagnostic Command Syntax	51
List command	51
TRACE command	52

Preface

This summary contains frequently used syntax information associated with Release 6.1.0 of the Host Software Component (MSP 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.

Related Documentation

The following list describes the publications that provide additional information about the HSC.

The online documentation is available at:

<http://docs.sun.com/app/docs/prod/stortek.hsc#hsc>

Function	Title
HSC Publications	
Tasks for planning and configuring the HSC	<i>Configuration Guide</i>
System operation, maintenance, and problem resolution	<i>Operator's Guide</i>
Detailed HSC and library information, control statements, and utilities	<i>System Programmer's Guide</i>
Messages and codes issued by the HSC	<i>Messages and Codes Guide</i>
Frequently used syntax information	<i>Reference Summary</i>
Related Publications	
Installation tasks for the HSC	<i>NCS Installation Guide</i>
Detailed descriptions of HSC user exits	<i>NCS User Exit Guide</i>
Interface between the HSC and the operating system that influences tape allocation	<i>Storage Management Component (SMC) Configuration and Administration Guide</i>
Generate XML output for VTCS and NCS commands and utilities	<i>NCS XML Guide</i>

Documentation, Support, and Training

Function	URL
Documentation	
• Customer:	http://docs.sun.com
• Employee:	http://docs.sfbay.sun.com/
• Partner:	https://spe.sun.com/spx/control/Login
Downloads	
• Customer	http://www.sun.com/download/index.jsp
• Employee	https://dlrequest-zn-dlapps1.sfbay.sun.com/usr/login
Support	http://www.sun.com/support/
Training	http://www.oracle.com/global/us/education/sun_select_country.html
Online Account	https://reg.sun.com/register

Oracle Welcomes Your Comments

Oracle is interested in improving its documentation and welcomes your comments and suggestions. Submit your comments by clicking the Feedback[+] link at:

<http://docs.sun.com>

Please include the title and part number of your document with your feedback:

Reference Summary, 312613801

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*) —►◀

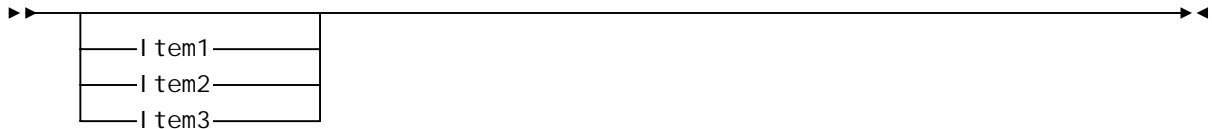
Single Required Choice

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.

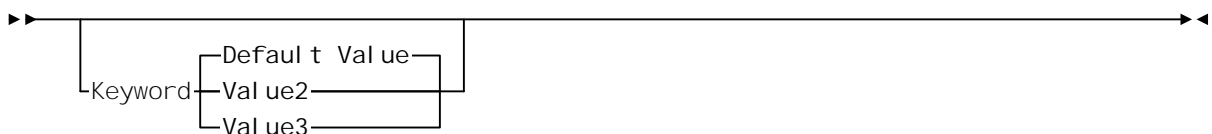


Defaults

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.

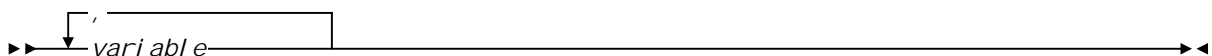


Some keyword parameters provide a choice of values in a stack. When the stack contains a default value, the keyword and the value choices are placed below the baseline to indicate that they are optional, and the default value appears above the keyword line. In the following example, if the keyword is 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 (*ll*) is a hexadecimal value from 00 through 17. 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 the *HSC/MSP Configuration Guide* for information about the SLIACS macro. The first LSM listed for an ACS acquires a hexadecimal number of 00, the second LSM listed for an ACS acquires a hexadecimal number of 01, and so forth.

- An LSMid (*lsm-id*) is made up of the ACSid and the LSM number separated by a colon (:). It differentiates an LSM from every other LSM in a library.
- A CAP number is a hexadecimal value from 00 to 02 for all existing libraries, except for SL3000 libraries. The CAP number for an SL3000 is a hexadecimal value from 00 to 0B. 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.

StreamLine Library (Model SL3000)

The library is configured with 2, 13-cell removable magazines in each CAP. This library can contain up to 12 CAPs (00-0B).

StreamLine Library (Model 8500)

This library is configured with 3, 13-cell removable magazines. An optional 39-cell CAP can be added.

CAPid Formats

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

- *AA:LL*, where *AA* is the ACSid and *LL* is the LSM number. This format is referred to as the *lsm-id*.
- *AA:LL:CC*, where *AA:LL* 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 MSP 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 *AA:LL* 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 *AA:LL: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 4410 or 9310 standard CAP
- the right-hand 40-cell 4410 or 9310 enhanced CAP
- the 20-cell 9360 WolfCreek CAP
- a 14-cell or 10-cell removable magazine 9740 TimberWolf CAP
- a left-side SL3000 AEM CAP, consisting of 18, 13-cell removable magazines
- the left-hand 39-cell SL8500 library CAP.

01 indicates one of the following:

- the left-hand 40-cell 4410 or 9310 enhanced CAP
- the 30-cell 9360 WolfCreek optional CAP
- an optional SL3000 CEM CAP consisting of 2, 13-cell removable magazines
- the right-hand optional 39-cell SL8500 library CAP.

02 indicates one of the following:

- PCAP in either a 4410 or 9310 enhanced CAP or a 9360 WolfCreek CAP.
- an optional SL3000 CEM CAP consisting of 2, 13-cell removable magazines.

03, 04, 05 indicates the following:

- an optional SL3000 CEM/DEM CAPs each consisting of 2, 13-cell removable magazines.

06 indicates the following:

- the only required SL3000 BDM CAP consisting of 2, 13-cell removable magazines.

07, 08, 09, 0A indicates the following:

- an optional SL3000 CEM CAPs each consisting of 2, 13-cell removable magazines.

0B indicates the following:

- a right-side SL3000 AEM CAP, consisting of 18, 13-cell removable magazines



Notes:

- If SL3000 library CAPs are not present, the HSC reports them as “not installed” This keeps CAP addresses from changing as new CAPs are added.
- Refer to the individual command explanations for additional information on specifying CAPids.
- SL3000 and SL8500 libraries do not contain a PCAP.

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, any portion is valid as the incremental range. For example:

A00B00 the largest range that can be specified is A00B00 through A99B99.

A0B0CC the largest range that can be specified is A0B0CC through A9B9CC.

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:

A00A0-A99A0 increments VOLSERs A00A0 through A09A0, then A10A0 through A99A0.

9AA9A-9ZZ9A increments VOLSERs 9AA9A through 9AZ9A, then 9BA9A through 9ZZ9A.

111AAA-111ZZZ increments VOLSERs 111AAA through 111AAZ, then 111ABA through 111ZZZ.

999AM8-999CM8 increments VOLSERs 999AM8 through 999AZ8, then 999BA8 through 999CM8

A3BZZ9-A3CDE9 increments VOLSERs A3BZZ9 through A3CAA9, then A3CAB9 through A3CDE9

AAAAAA-AAACCC increments VOLSERs AAAAAA through AAAAAZ, then AAAABA through AAACCC

CCCNNN-DDDNNN 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
AAAAA-ZZZZZ	26^5	11,881,376
AAAAAA-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:

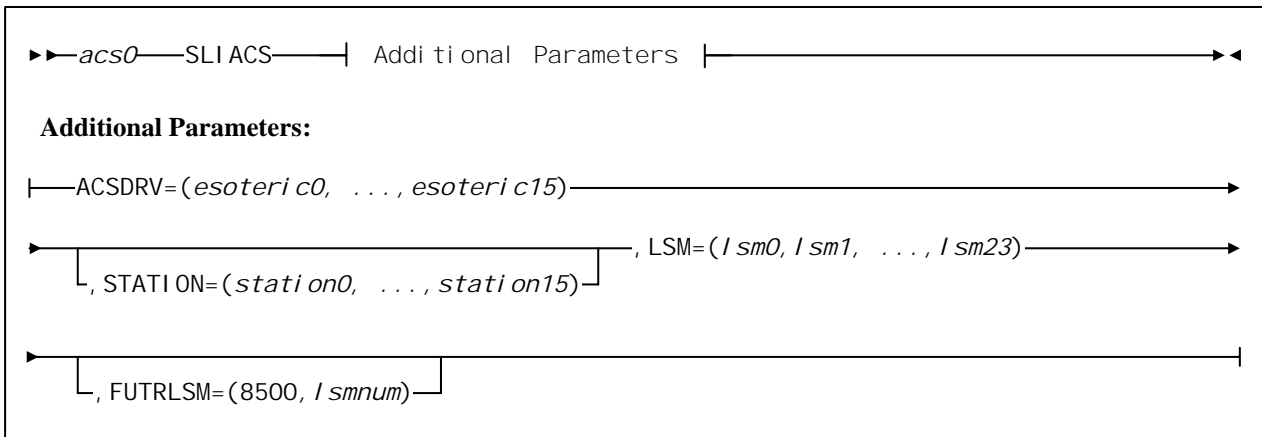
```
SCRPOOL NAME=STD36, RANGE (AAA000-AAA999, ZZZ+  
000-ZZZ999)
```

- 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.

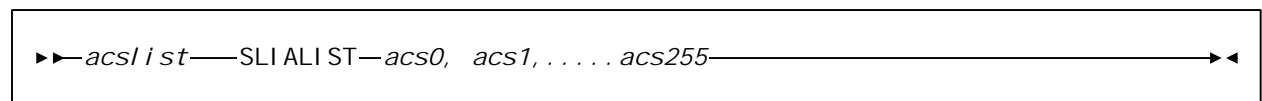
LIBGEN Macro Syntax

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/MSP Configuration Guide*.

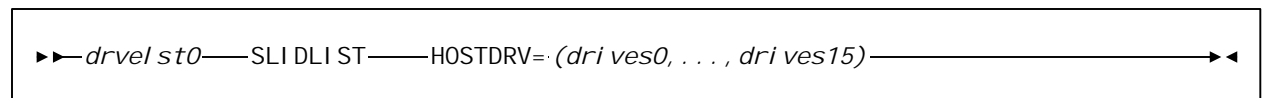
SLIACS macro



SLIALIST macro



SLIDLIST macro



SLIDRIVS macro

►► *dri ves0* SLI DRI VS ADDRESS=(*addr0, addr1. . .*) ◀◀

SLIENDGN macro

►► SLI ENDGN ◀◀

SLILIBRY macro

►► SLI LI BRY—HOSTI D=(*host-i d0, host-i d1, . . . , host-i d15*)—, ACSLI ST=*acsl i st*—

◀ Optional Parameters ▶

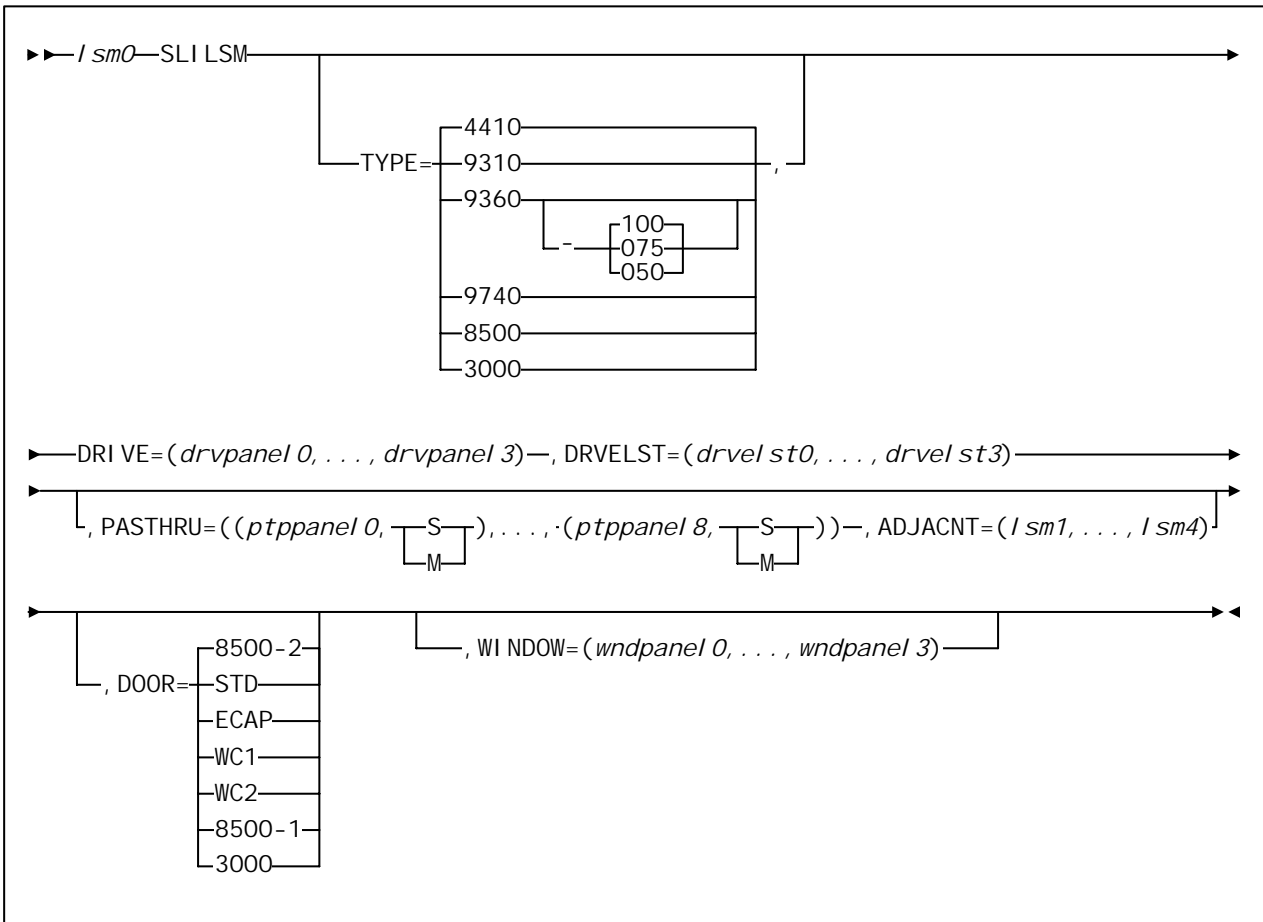
Optional Parameters:

◀, CLNPRFX=*CLN* *prefix* ◀, SMF=*255* *libtype* ▶

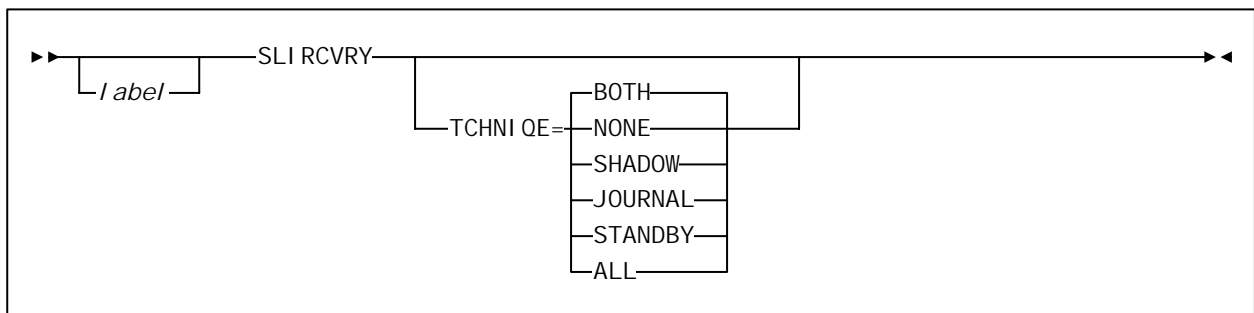
◀, MAJNAME=*STKALSQN* *qname* ◀, COMPRFX=*.* *commandchar* ◀, SCRLABL=*SL* *AL* *NL* *NSL* ▶

◀, EJCTPAS=*password* ◀, DRVHOST=(*host-i d*) ◀, FUTRACS=(*x, y*) ▶

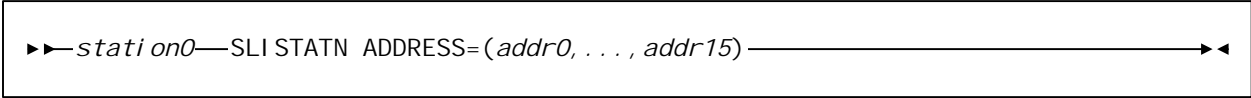
SLILSM macro

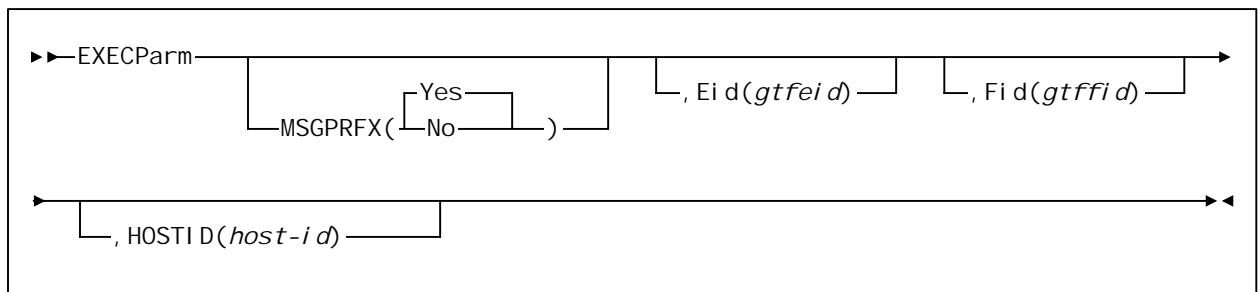
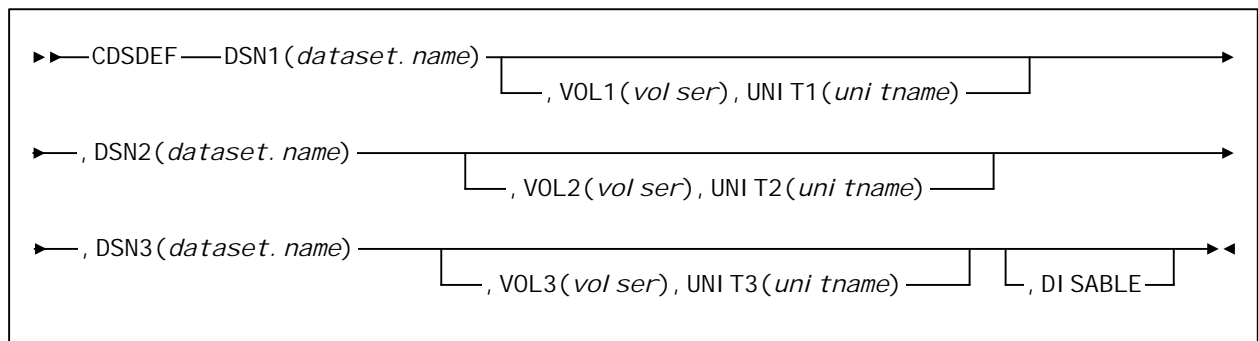


SLIRCVRY macro

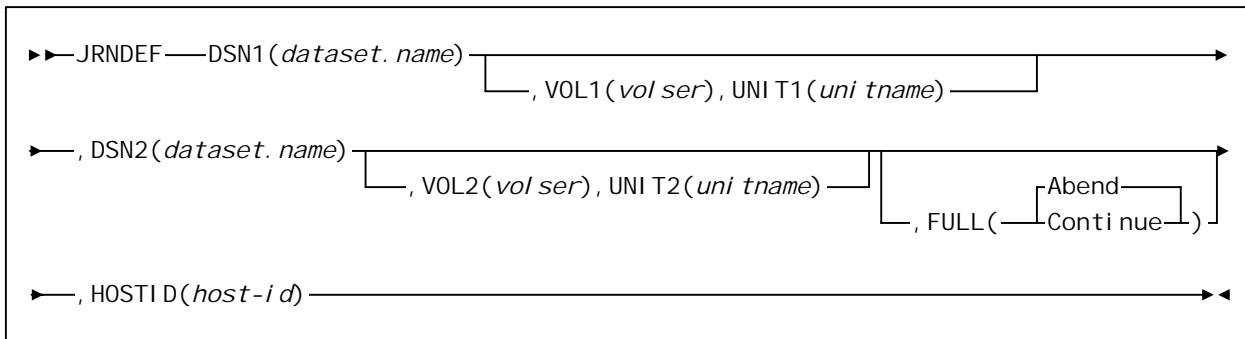


SLISTATN macro

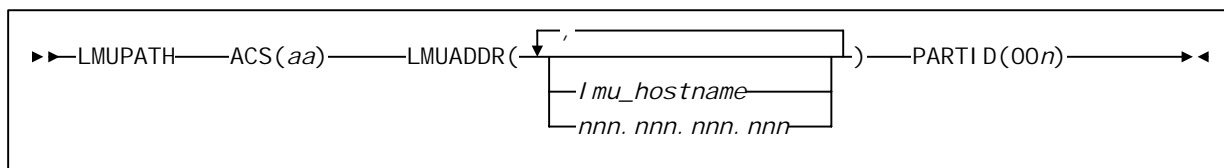




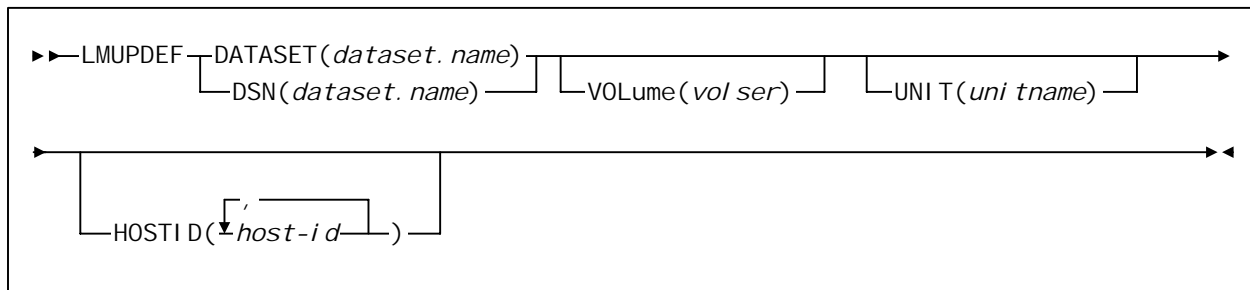
Journal Definition (JRNDEF) control statement



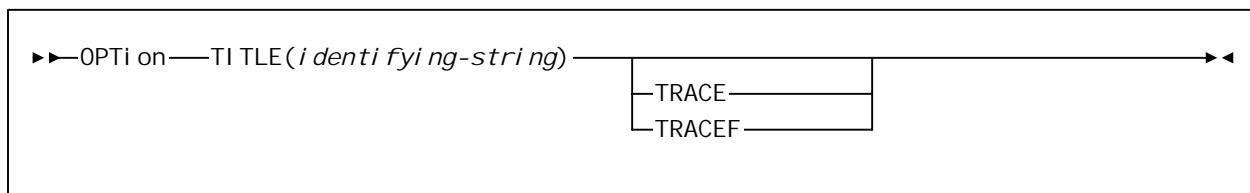
LMUPATH control statement



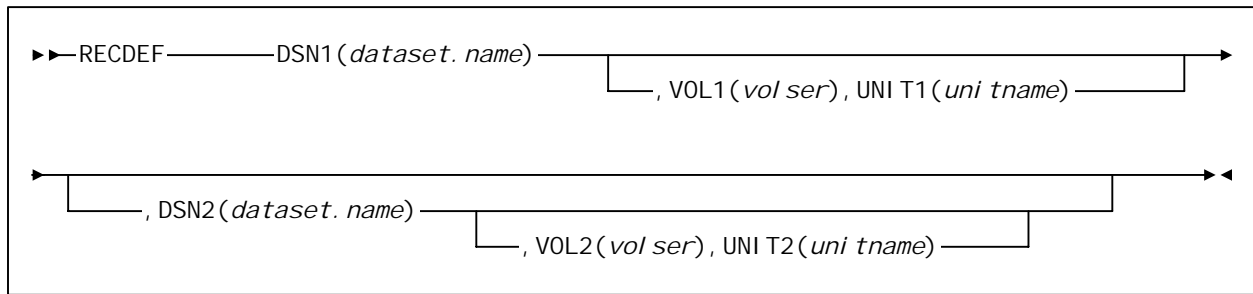
LMUPDEF command and control statement



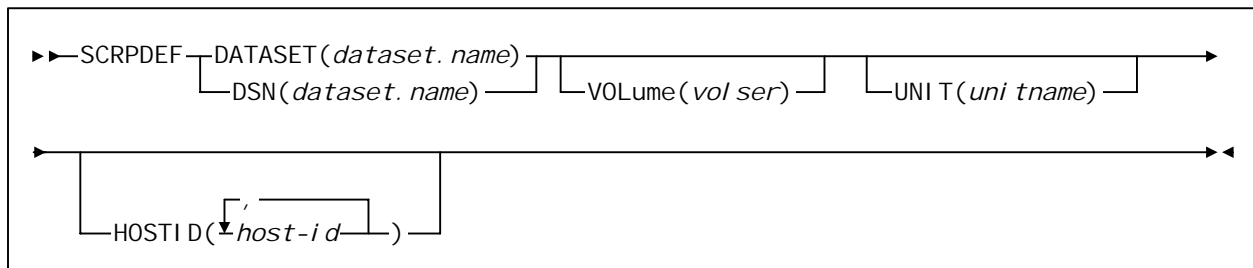
OPTion control statement



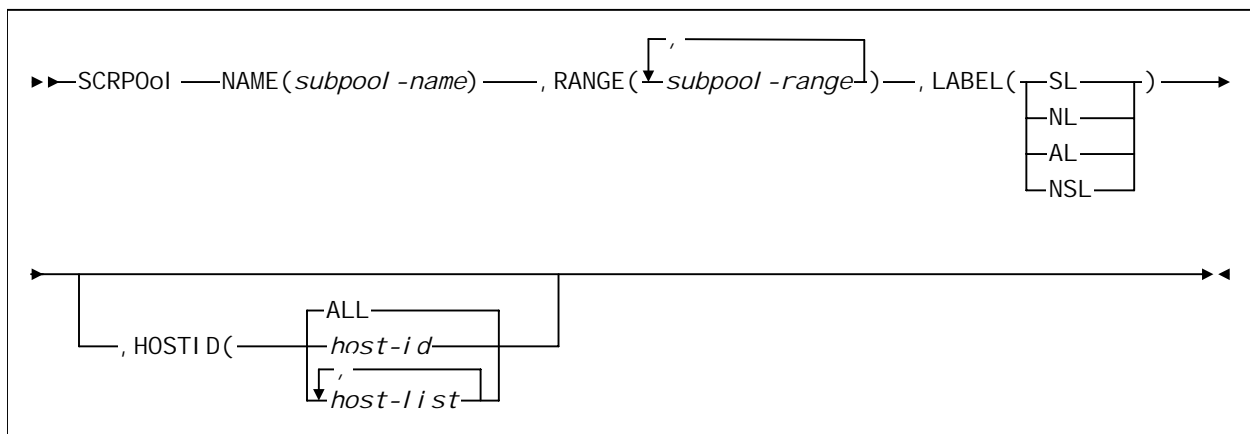
Reconfiguration Definition (RECDEF) control statement



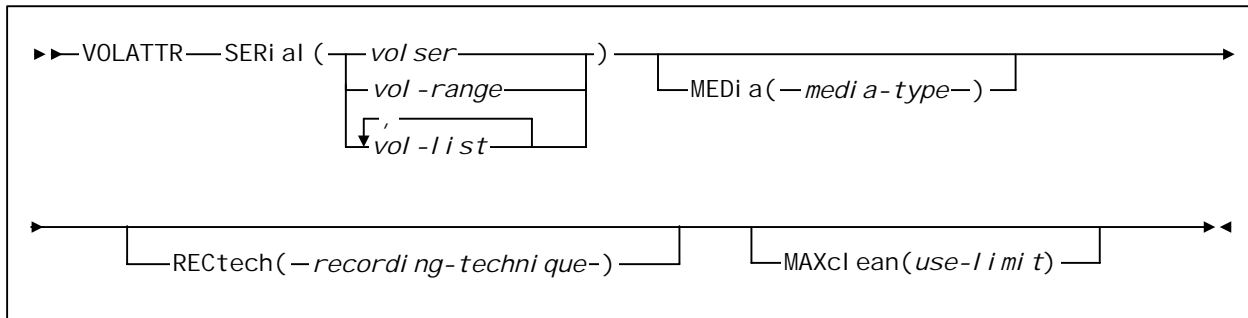
Scratch Subpool Definition (SCRPDDEF) command and control statement



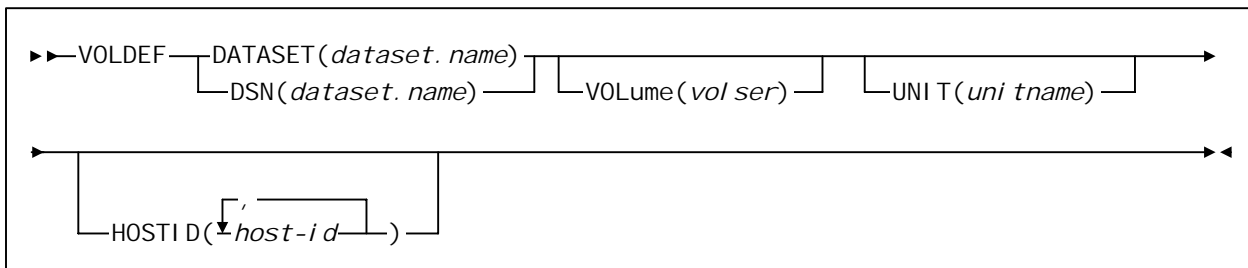
Scratch Subpool (SCRPOOL) control statement



Volume Attribute (VOLATTR) control statement



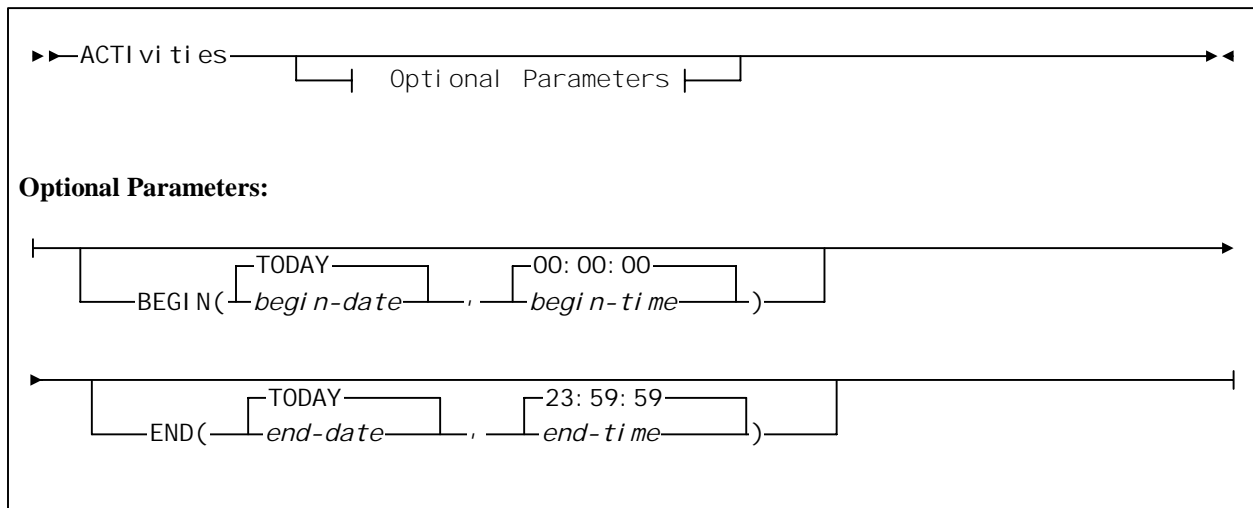
Volume Attribute Definition (VOLDEF) command and control statement



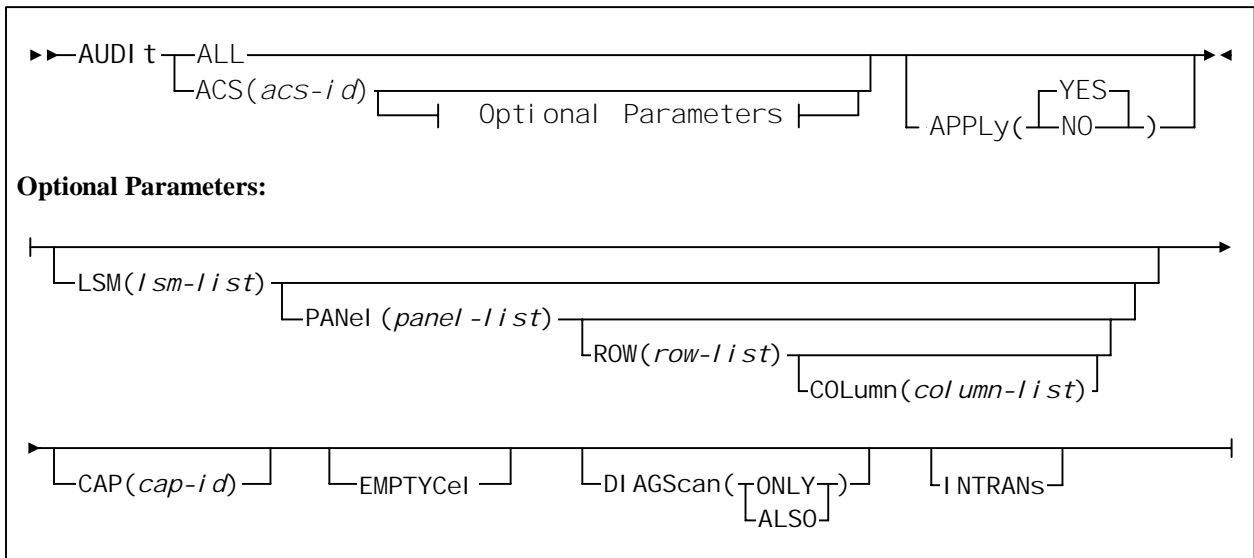
Utility Syntax

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

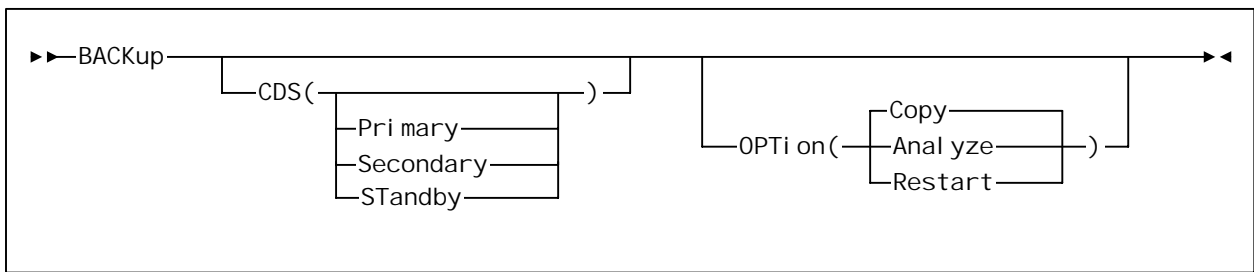
ACTivities Report utility



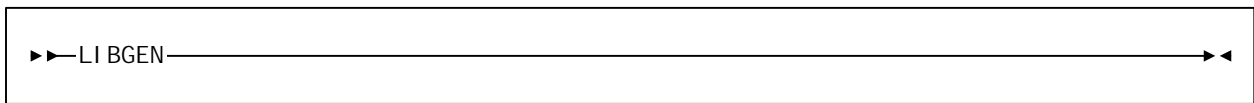
AUDIt utility



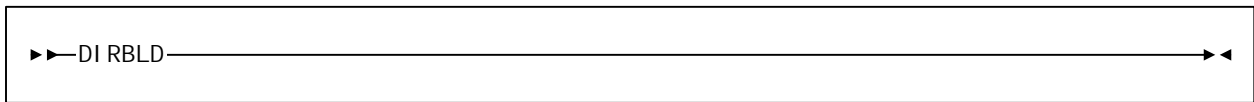
BACKUp utility



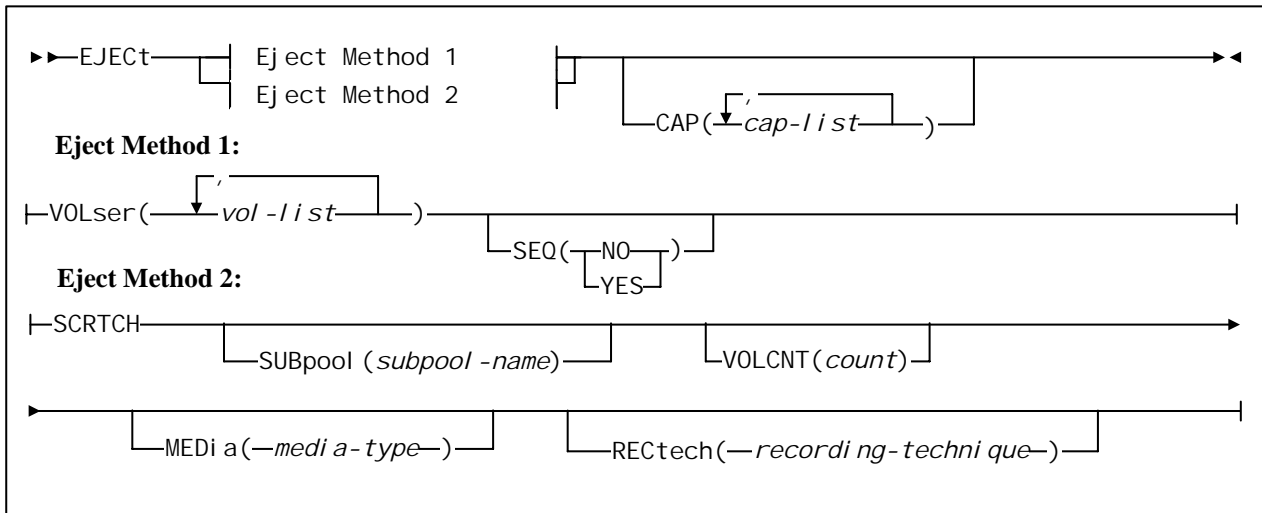
Database Decompile (LIBGEN) utility



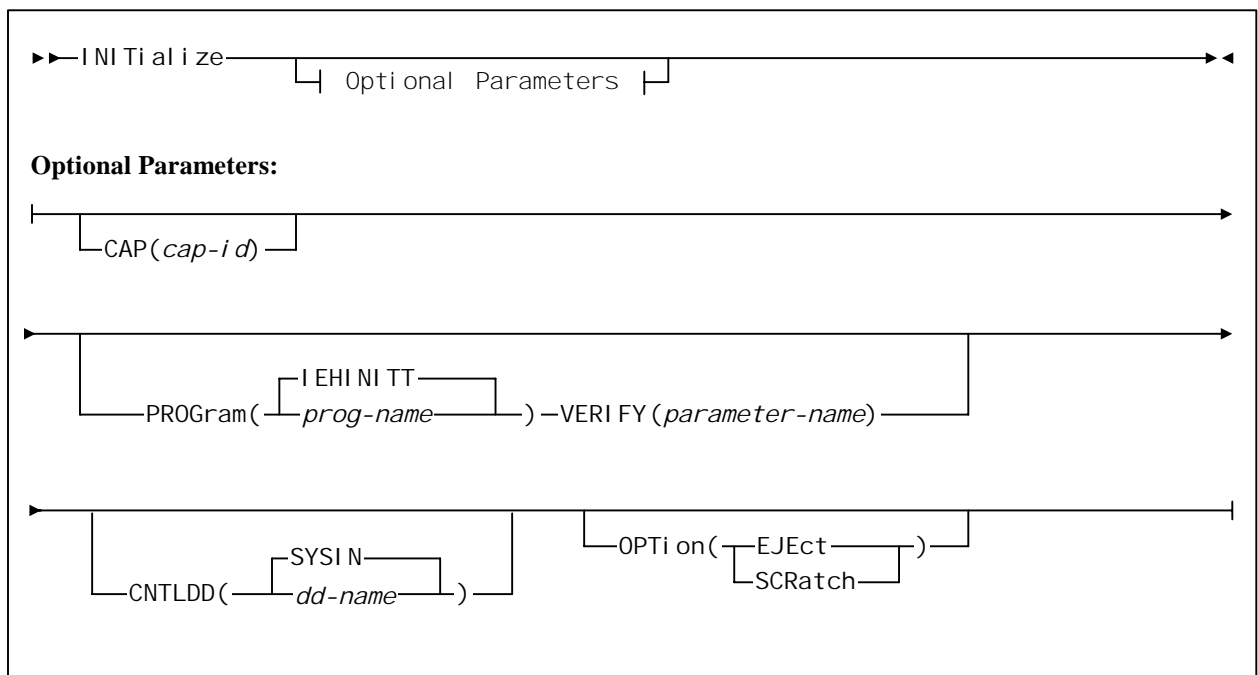
Directory Rebuild (DIRBLD) utility



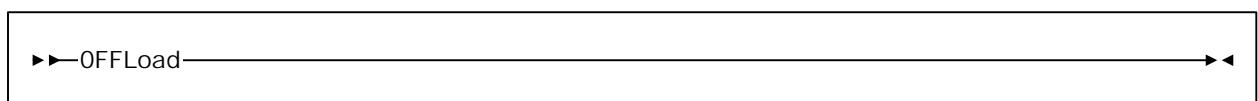
EJECT utility



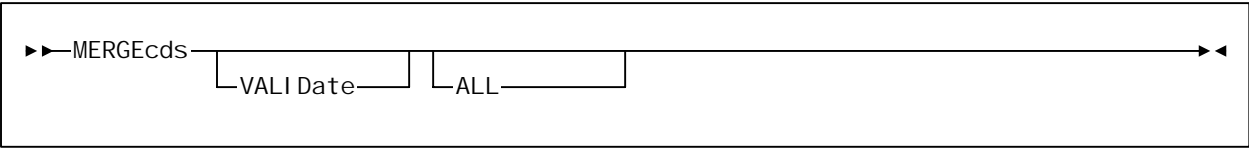
INITialize Cartridge utility



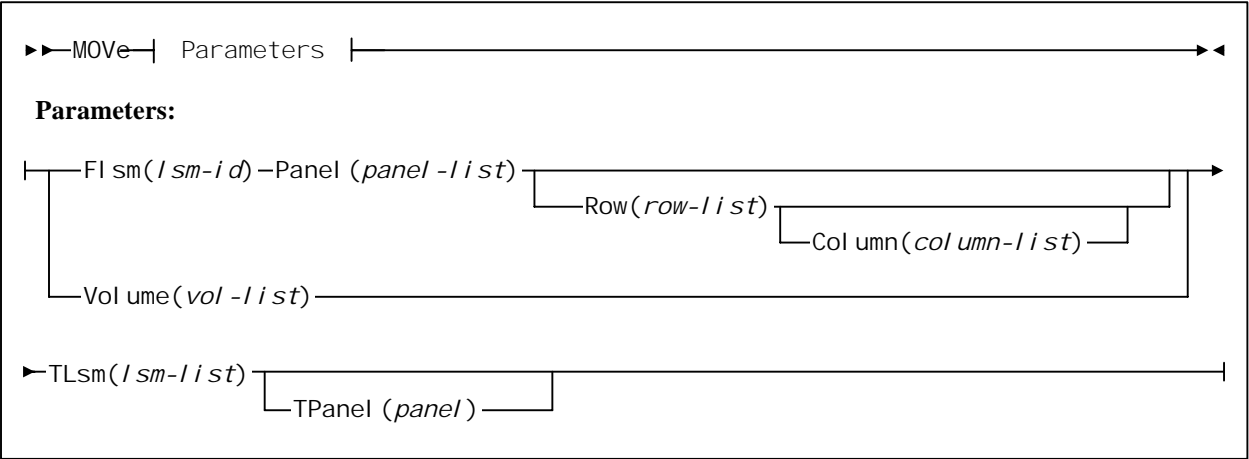
Journal OFFLoad utility



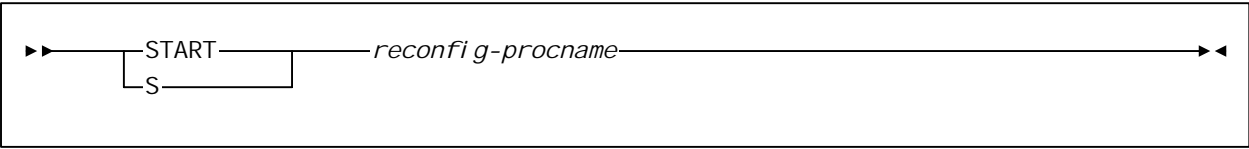
MERGEcds utility



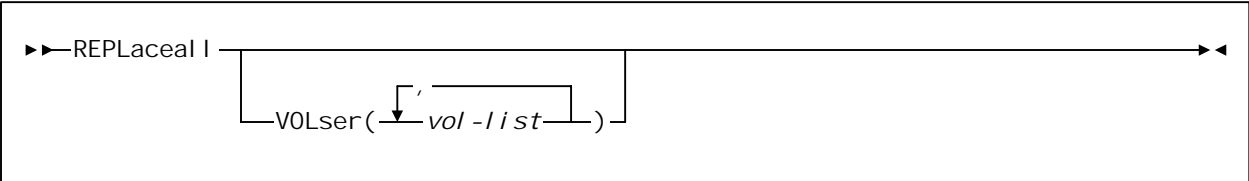
MOVE utility



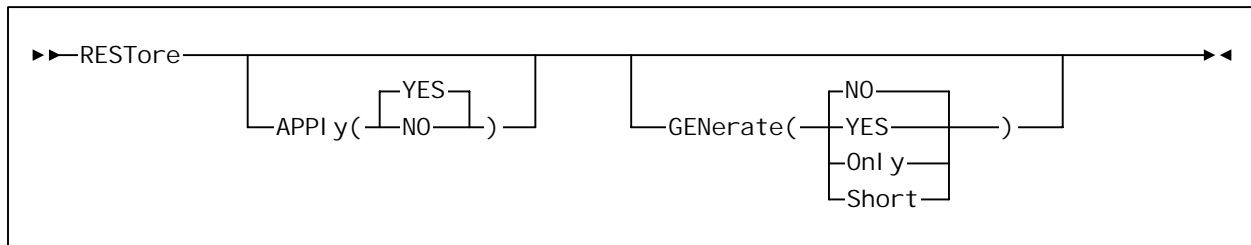
Reconfiguration utility



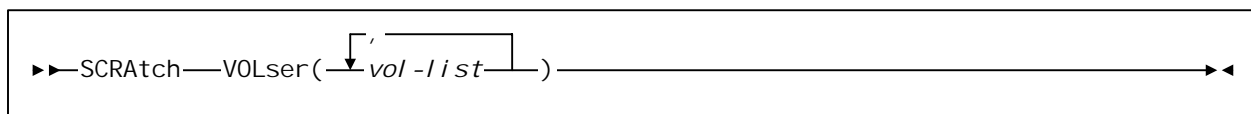
REPLace utility



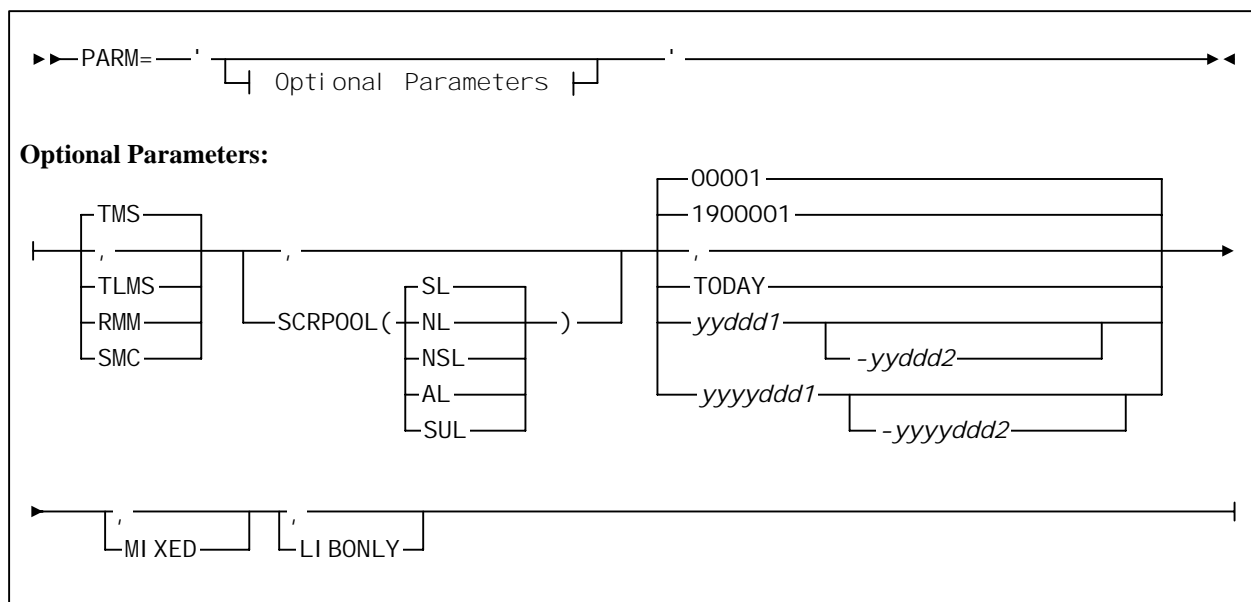
REStore utility



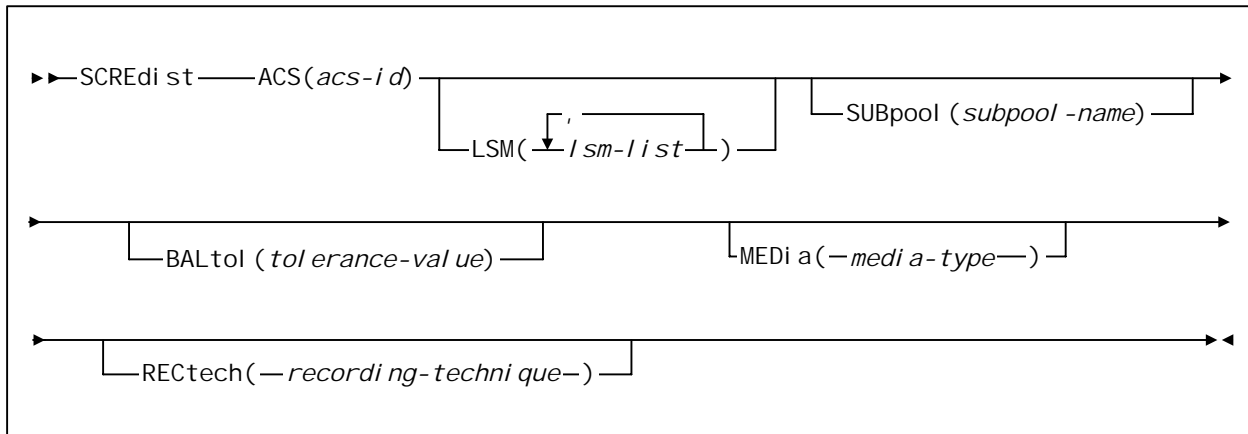
SCRAtch utility



Scratch Conversion utility



Scratch Redistribution (SCREdist) utility



SET utility

►►SET| Options |◀◀

Options:

|ACSDRV(*esoteric*)|, FORACS(*acs-id*)|
|, FORHOST(*host-id*)|

|CLNPRFX(*prefix*)|

|COMPRFX(*cmdhex*)|

|DELDI SP(|SCRTCH|)|
|NOSCRTCH|

|DRVHOST(|OFF|)|
|*host-id*|

|EJCTPAS(|
|*newpswd*|)|, OLDPASS(*oldpswd*)|

|FREEZE(|ON|)|, FORLSMID(*lsm-id*)|
|OFF|)|, 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)

```

▶▶SET—| Options |—————▶▶

Options:

|SLIDRVS(———), FORLSMID(lsm-id), FORPANEL(panel)——▶
      |——addr0——|
      |——, ... addr31——|

▶——|
  |——, FORHOST(host-id)——|

|SLI STATN(———), FORACS(acs-id)——|
      |——stat1, ..., stat16——|
      |——, FORHOST(host-id)——|

|SMF(libtype)——|

|TCHNIOE(———)——|
      |——NONE——|
      |——JOURNAL——|
      |——SHADOW——|
      |——BOTH——|
      |——STANDBY——|
      |——ALL——|

```

UNSCratch utility

```

▶▶UNSCratch—VOLser(———vol-list——)——▶▶

```

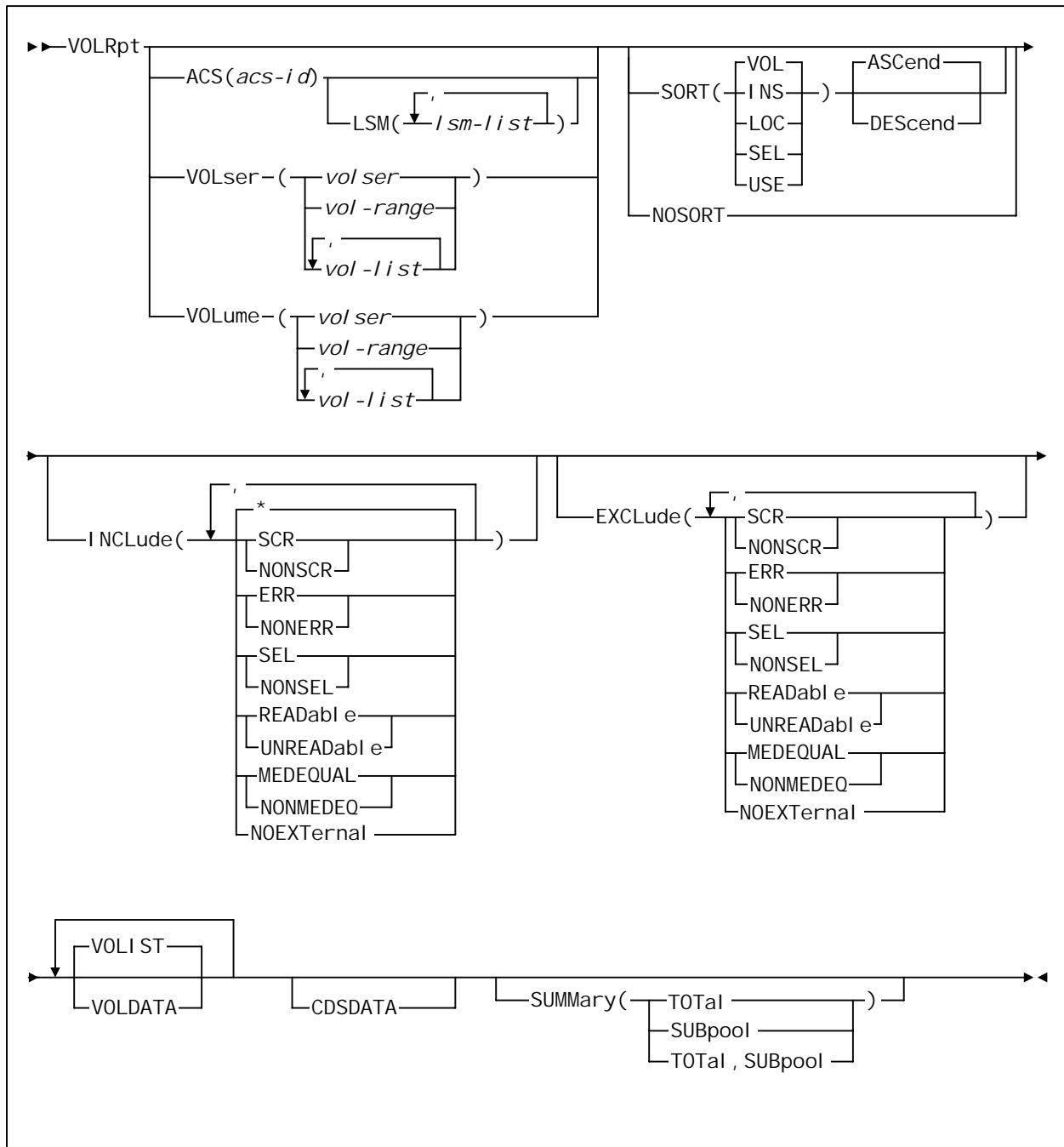
UNSElect utility

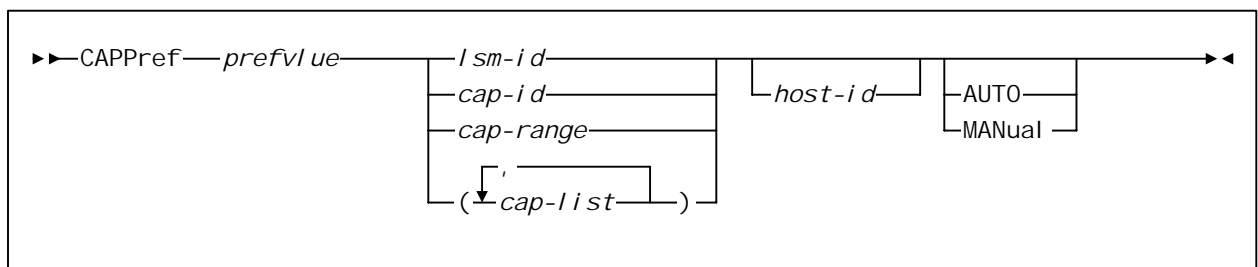
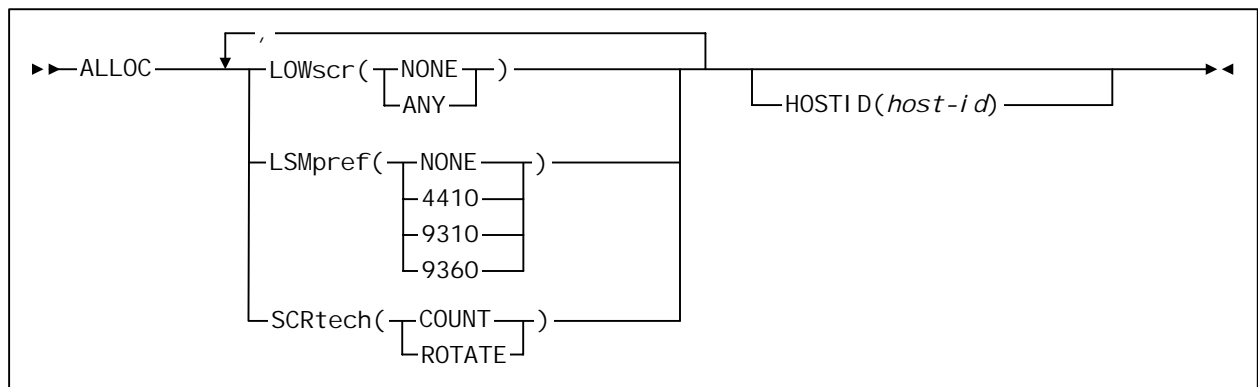
```

▶▶UNSElect—VOLser(vol ser)——|——▶▶
      |——, FORCE——|

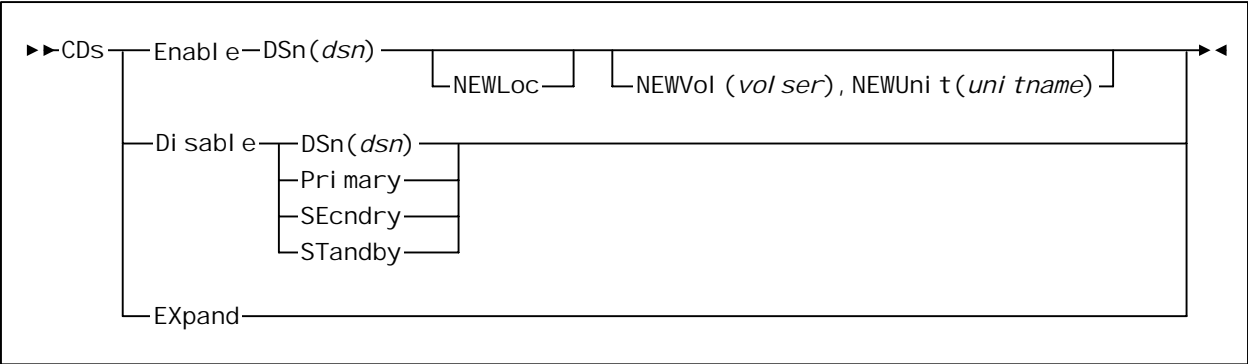
```

Volume Report (VOLRpt) utility

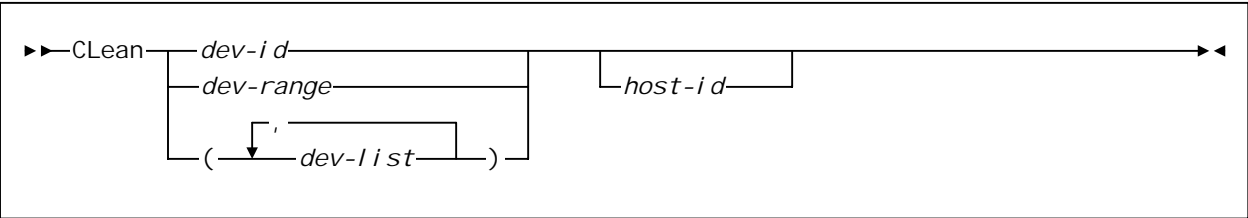




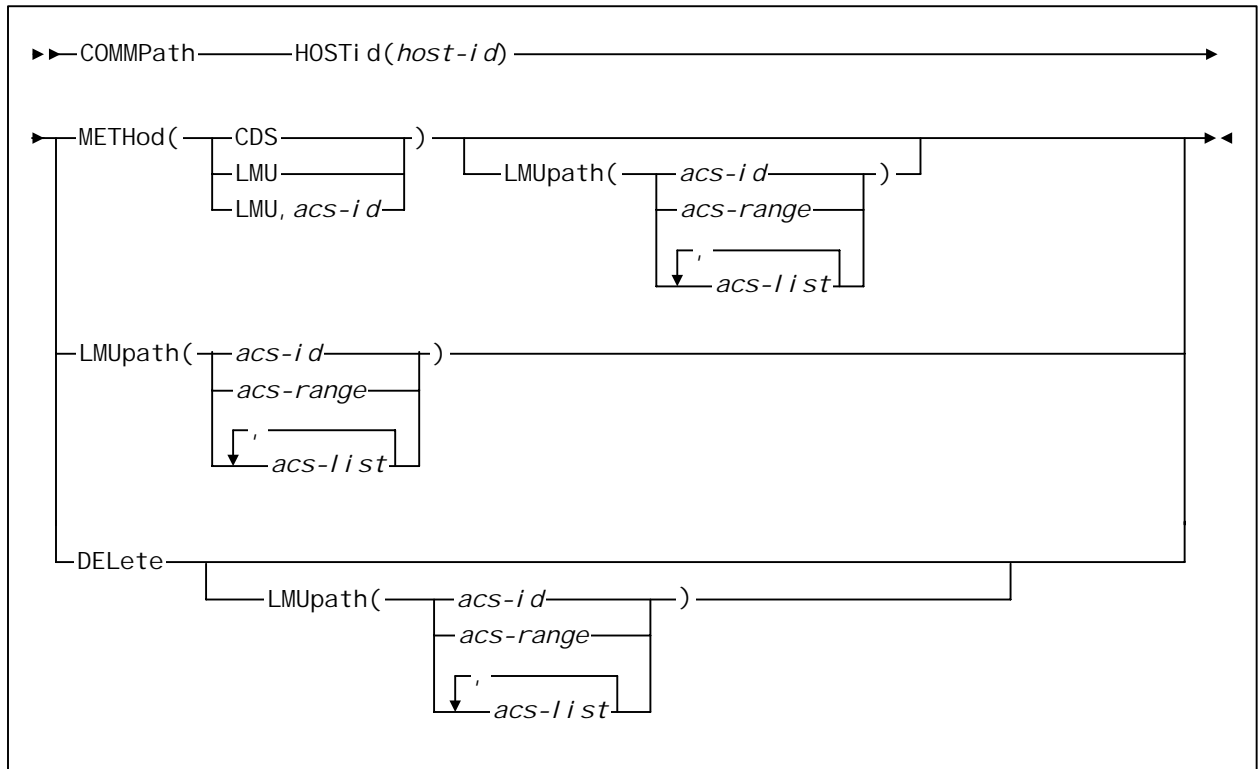
CDs Enable/Disable command



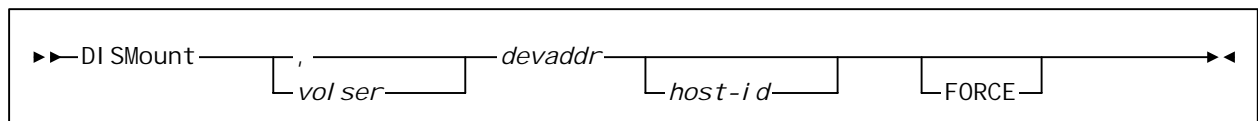
CLean command



Communications Path (COMMPath) command and control statement

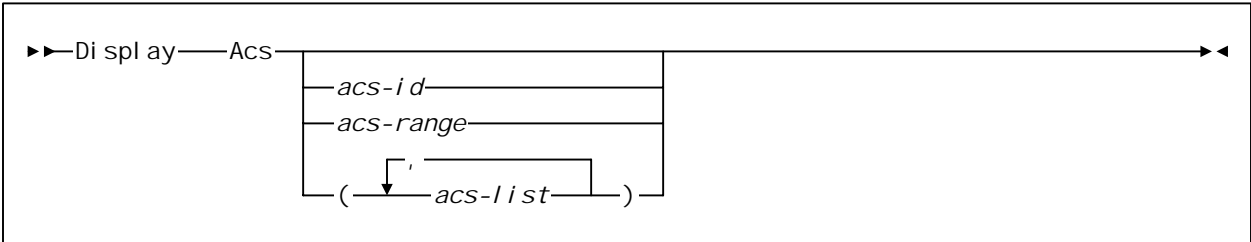


DISMount command

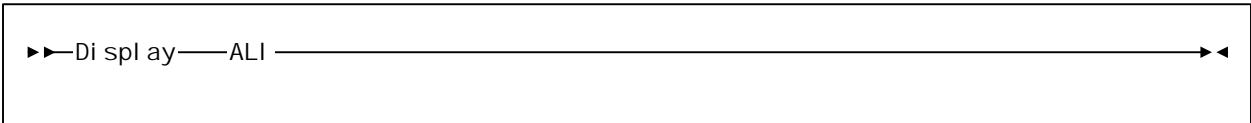


Display command

Display Acs



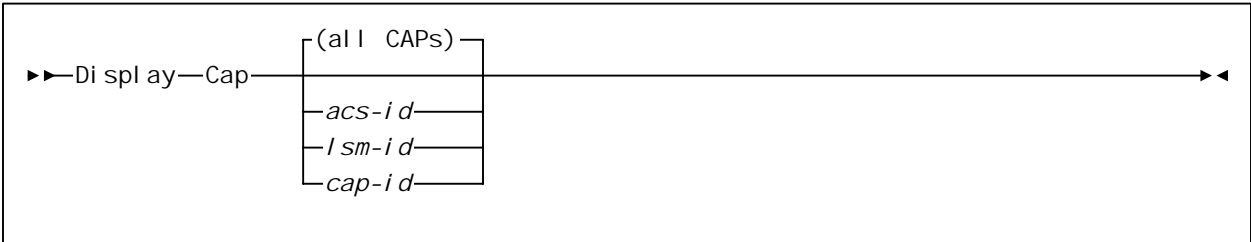
Display ALI



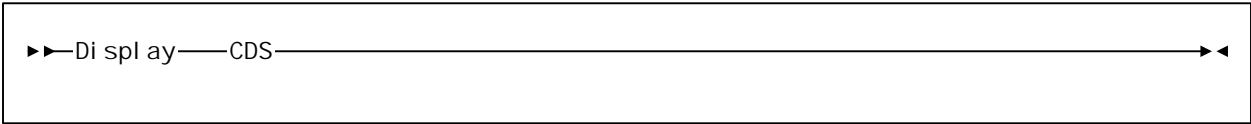
Display ALLOC



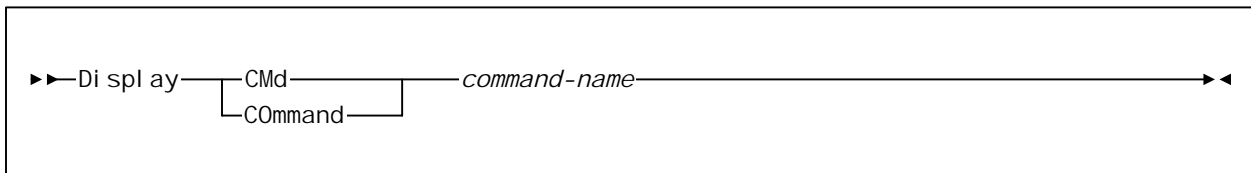
Display Cap



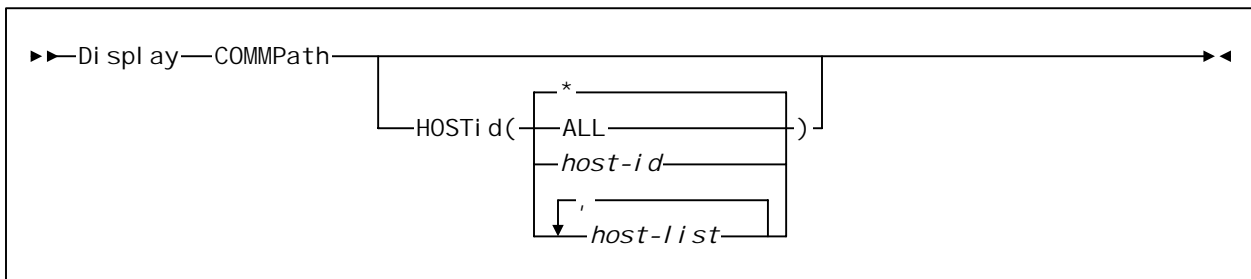
Display CDS



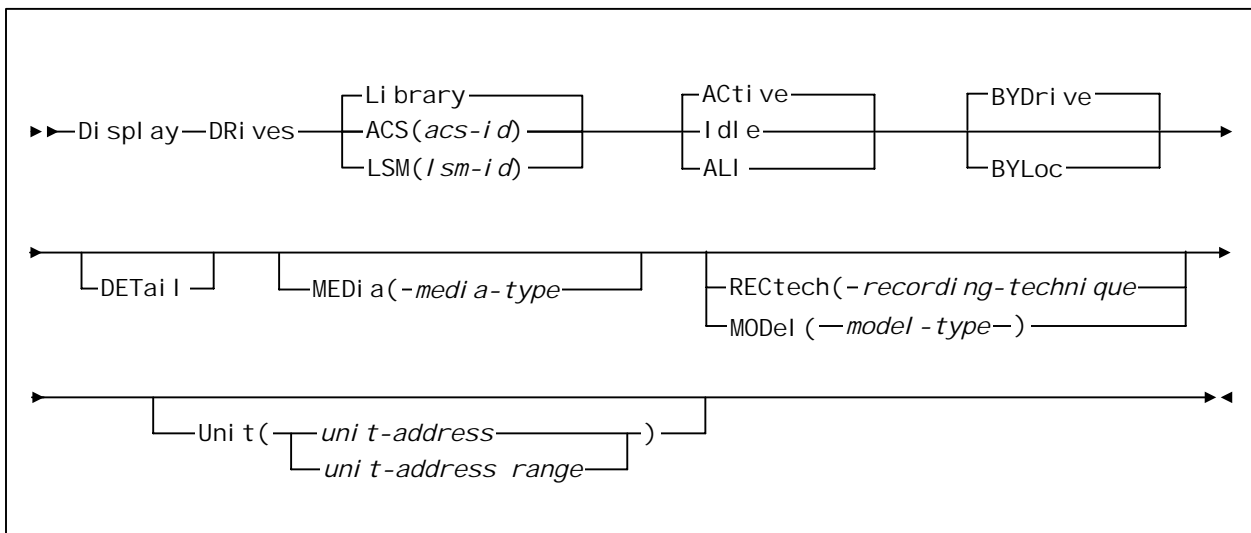
Display Cmd



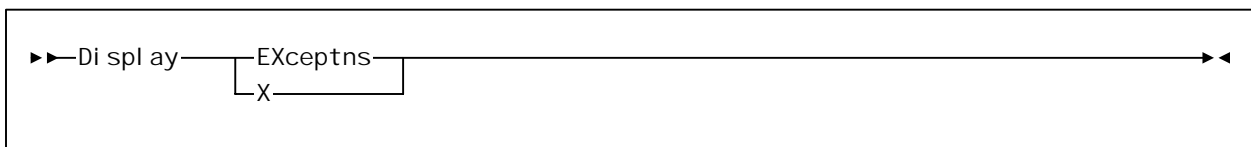
Display COMMPath



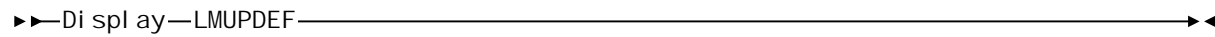
Display DRives



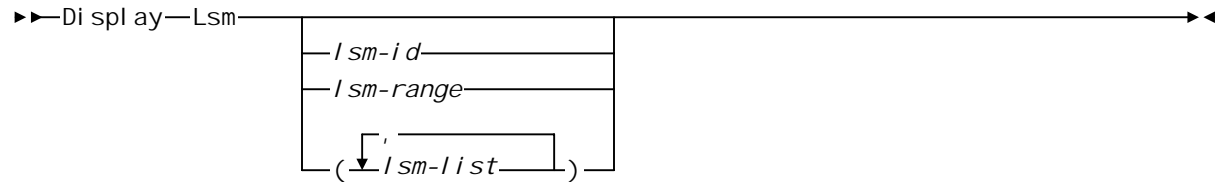
Display Exceptions



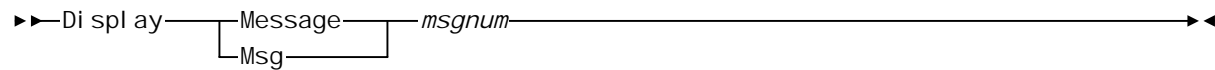
Display LMUPDEF



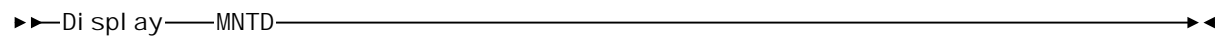
Display LSM



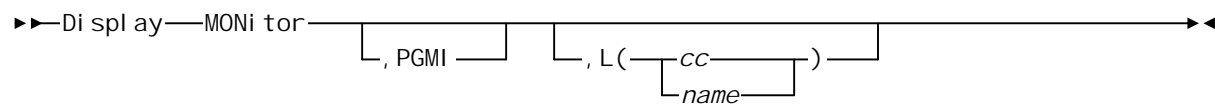
Display Message



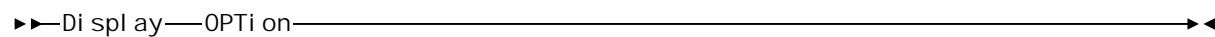
Display MNTD



Display MONitor



Display OPTion



Display Requests

►► Display — Requests ————— ►◀

Display SCRatch

►► Display — SCRatch ————— ►◀
 ┌ *acs-id* ─ SUBpool (*subpool -name*) ─ DETail ─
 └ *lsm-id* ─
┌ MEDia (— *media-type* —) ─ ─ RECtech (— *recordi ng-techni que* —) ─ ►◀

Display SCRPDEF

►► Display — SCRPDEF ————— ►◀

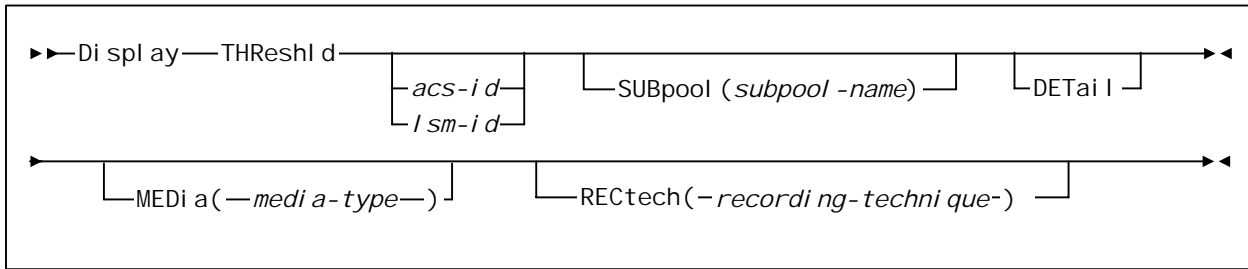
Display SRVlev

►► Display — SRVlev ————— ►◀

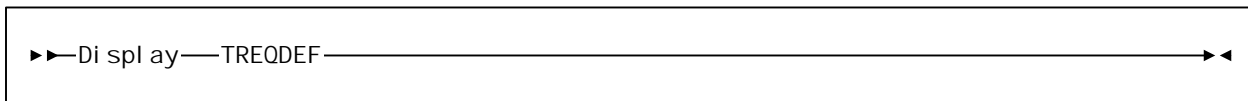
Display Status

►► Display — Status ————— ►◀

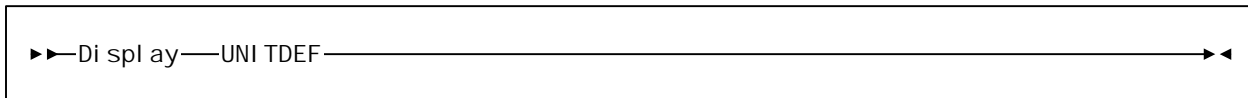
Display THResId



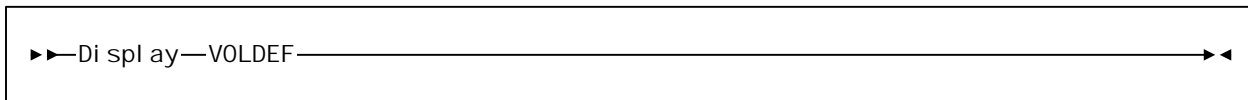
Display TREQDEF



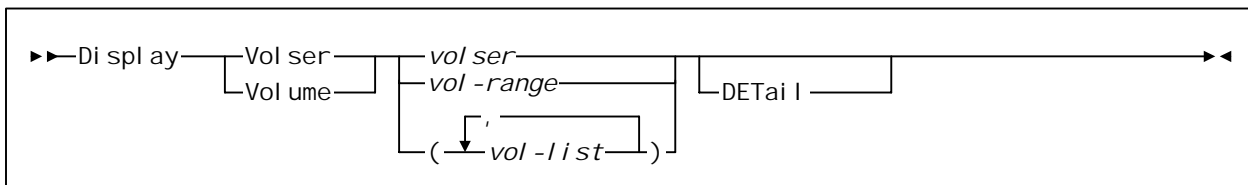
Display UNITDEF



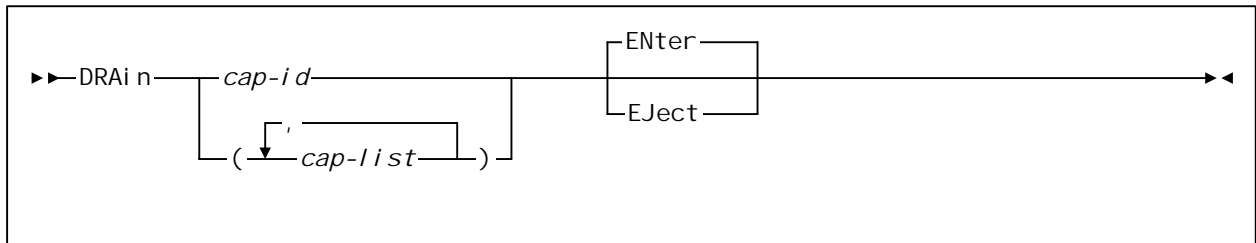
Display VOLDEF



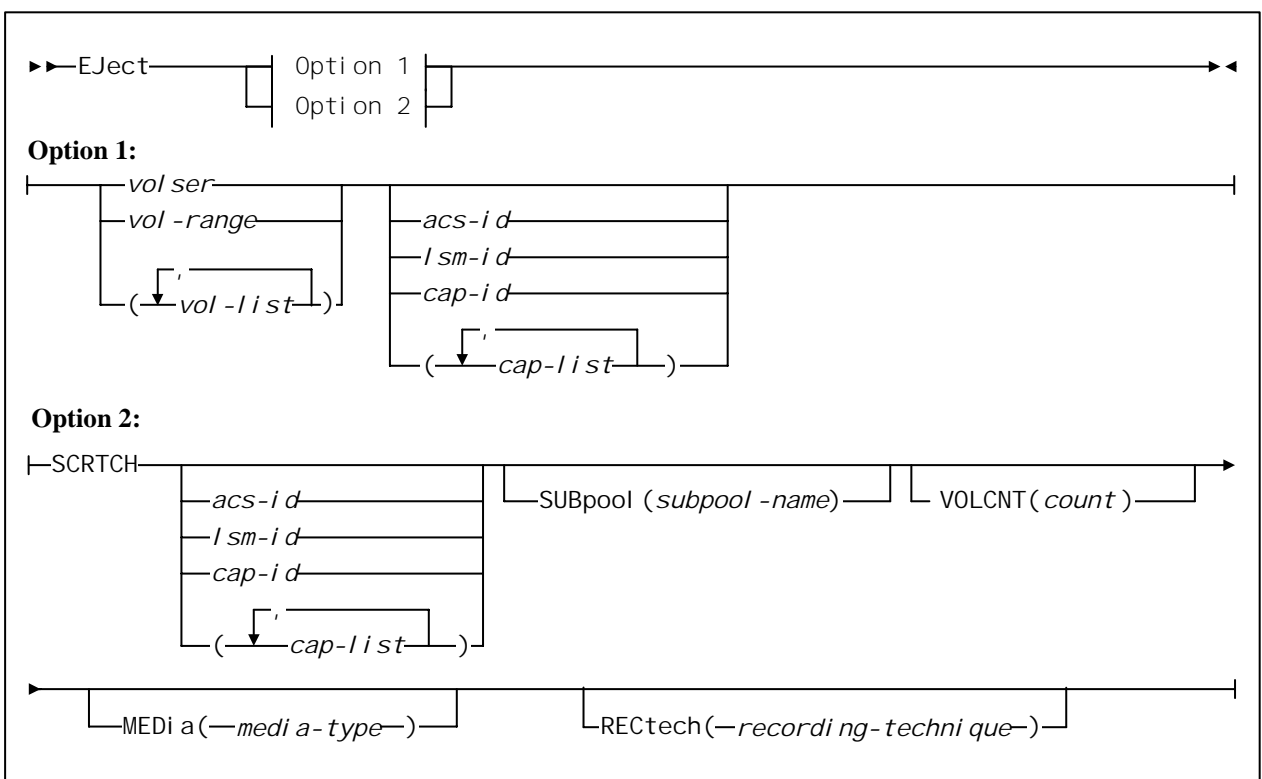
Display Volume



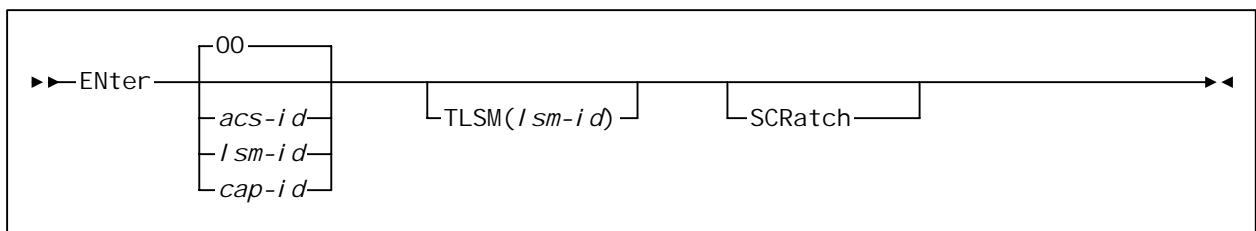
DRAIn CAP command



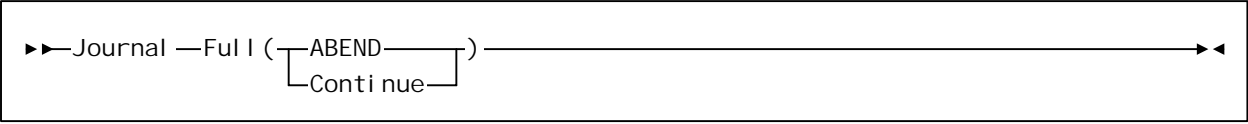
EJect command



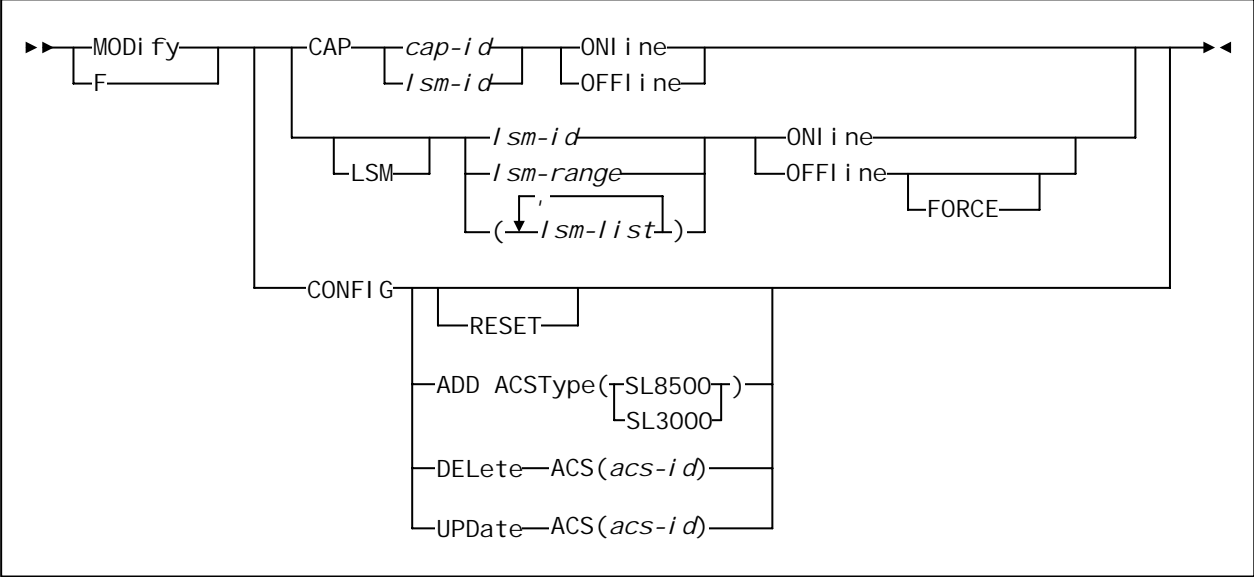
ENter command



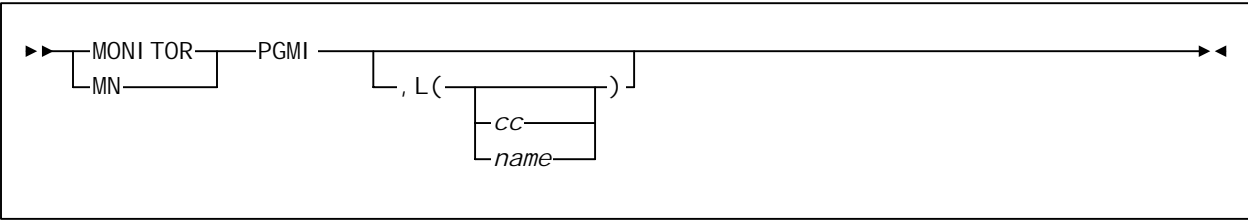
Journal command



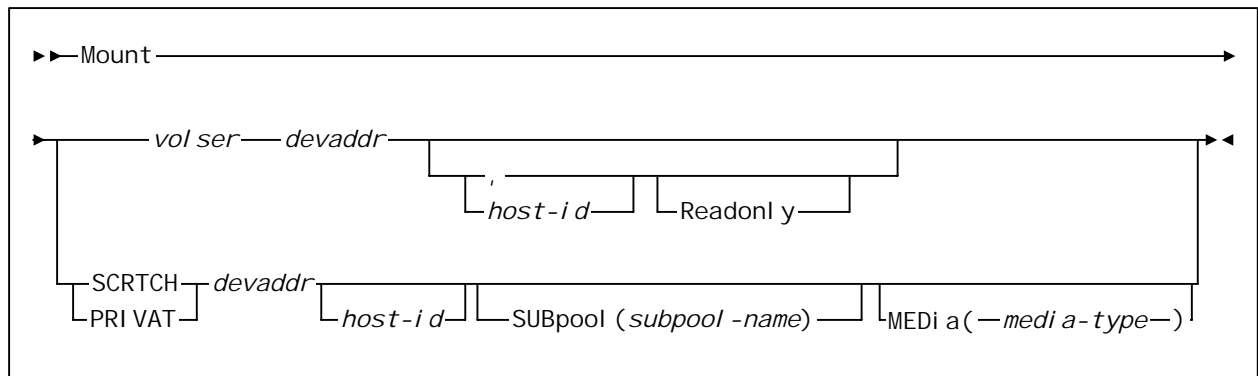
MODify command



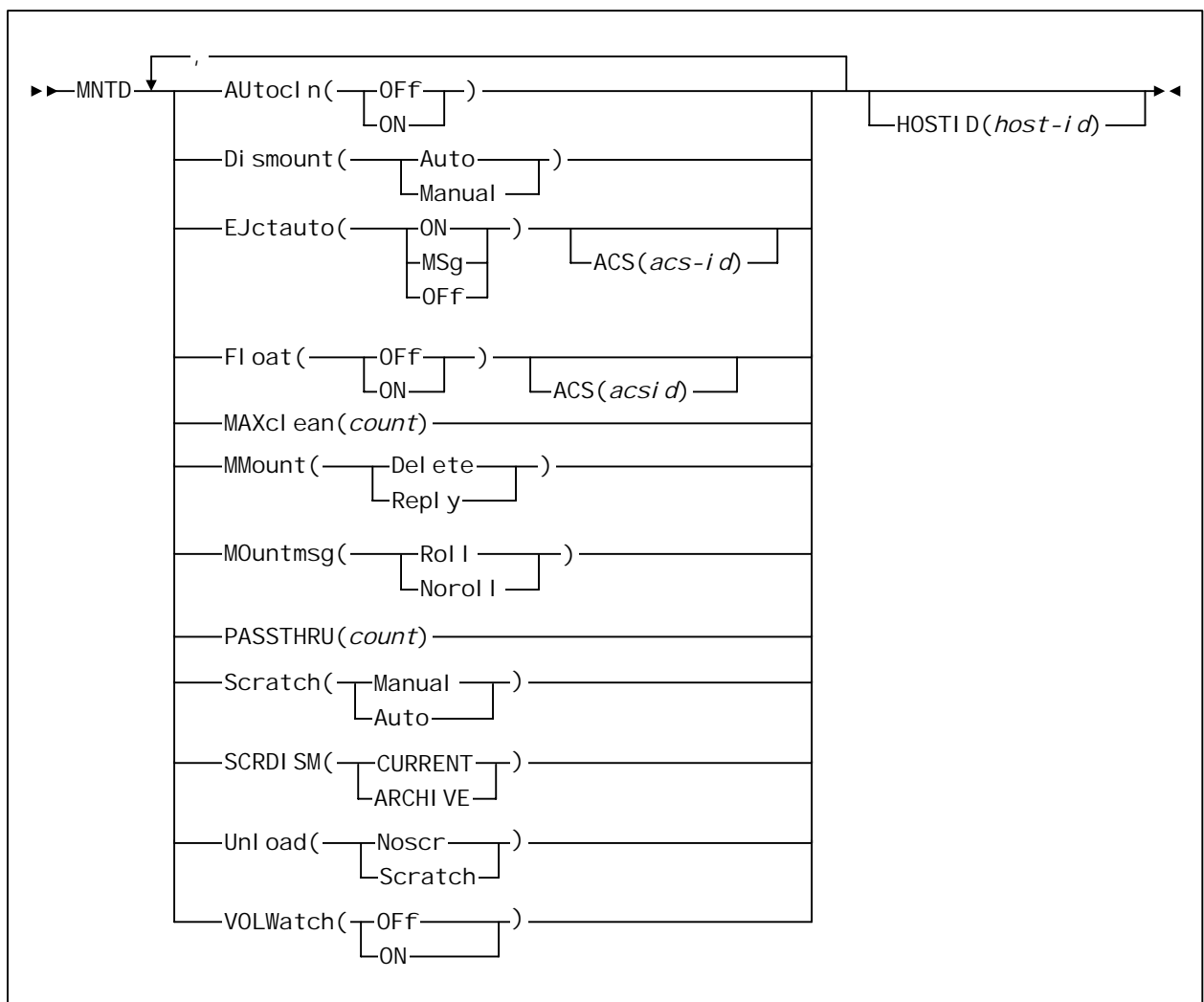
MONITOR command



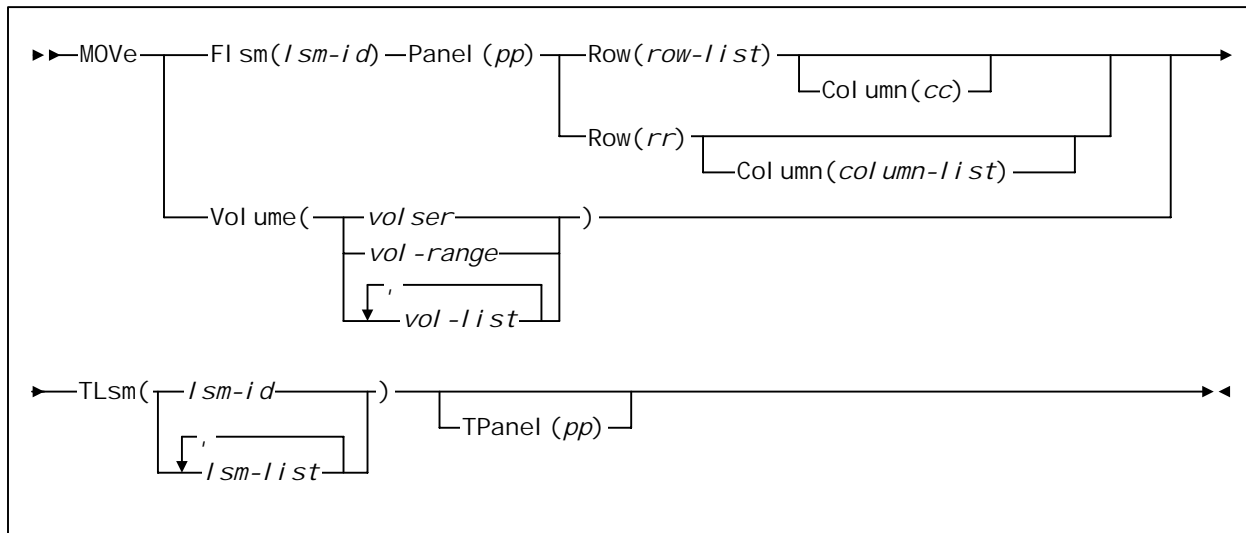
Mount command



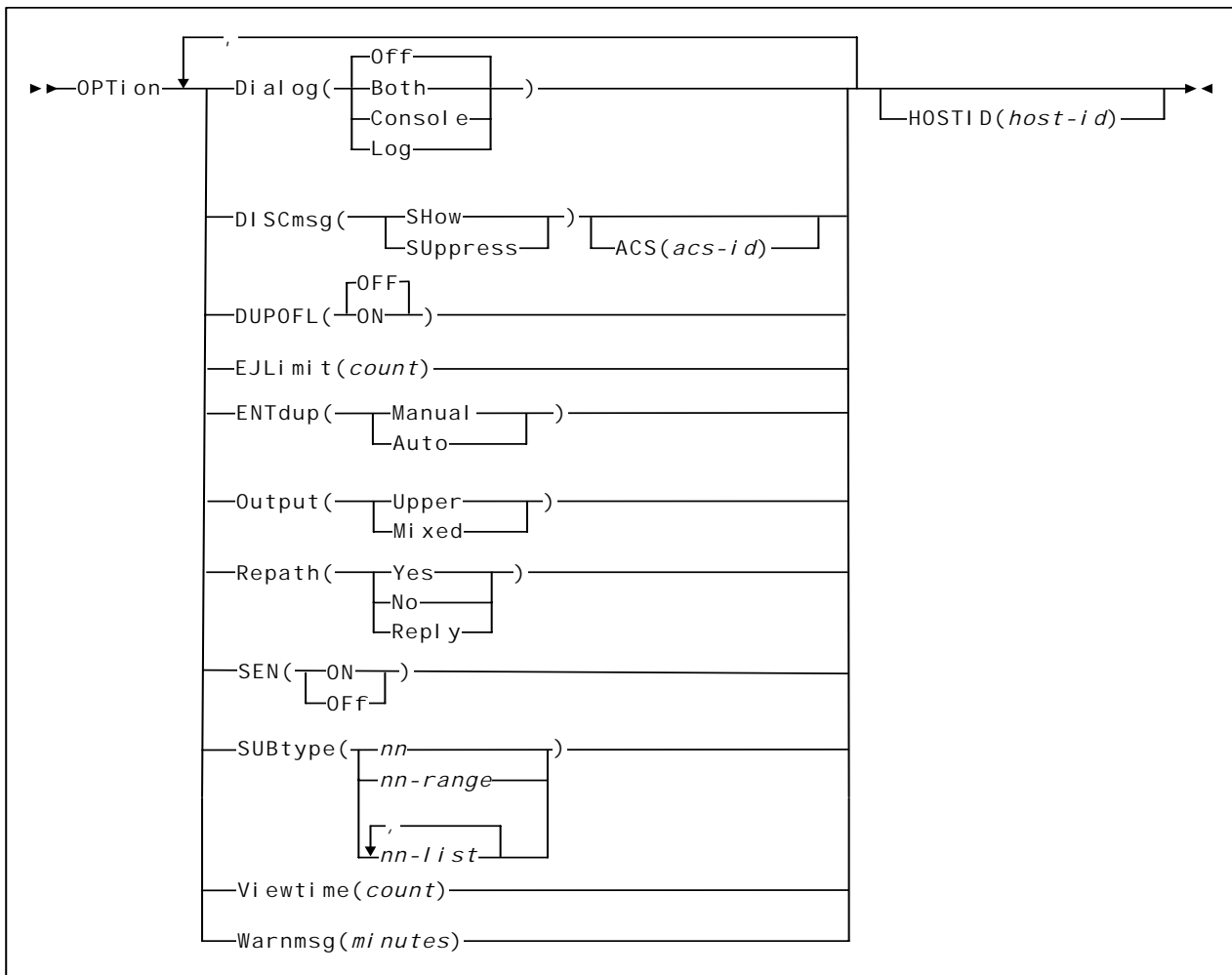
Mount/Dismount Options (MNTD) command and control statement



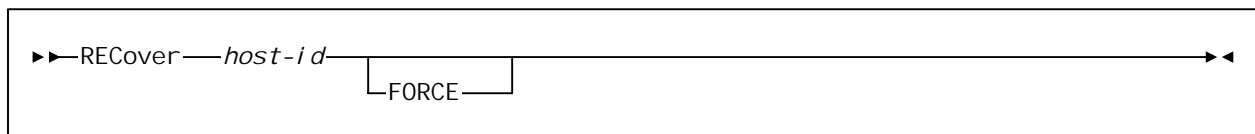
MOVE command



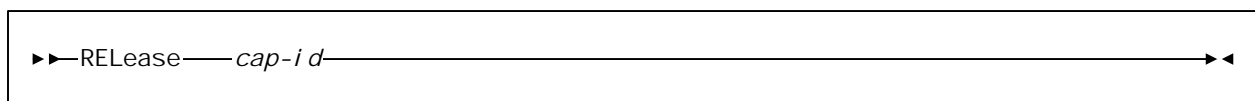
OPTion command and control statement



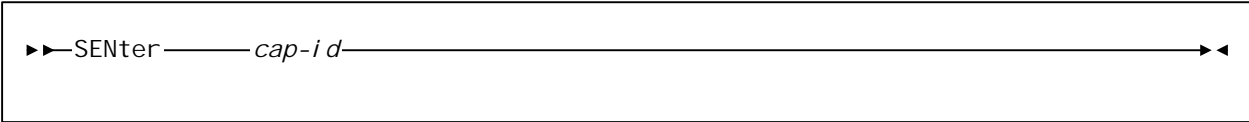
RECover Host command



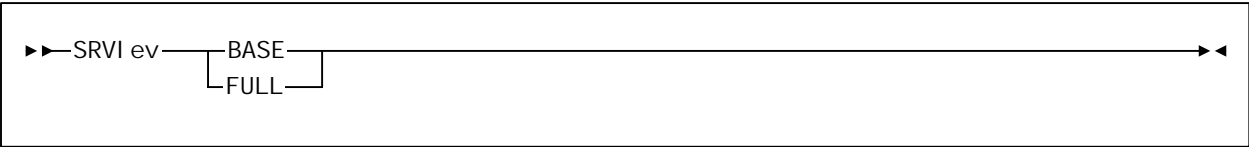
RELease CAP command



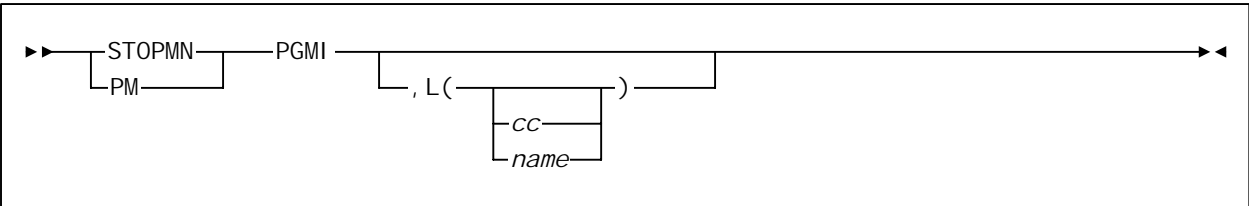
SEnTer command



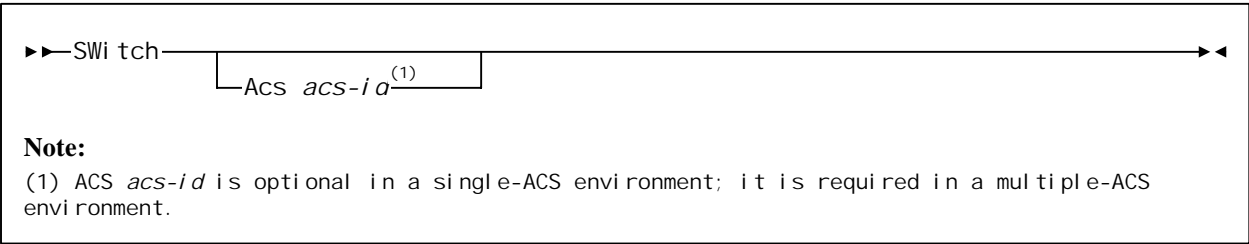
SRVlev command



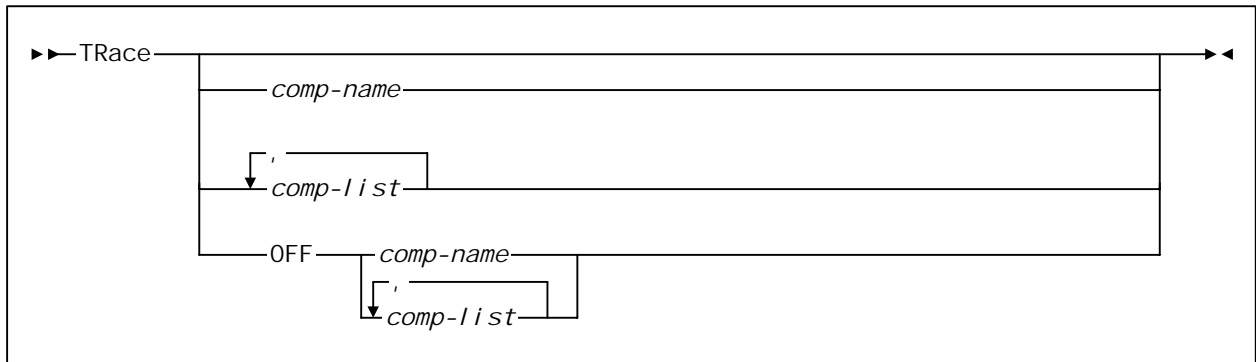
Stop Monitoring (STOPMN) command



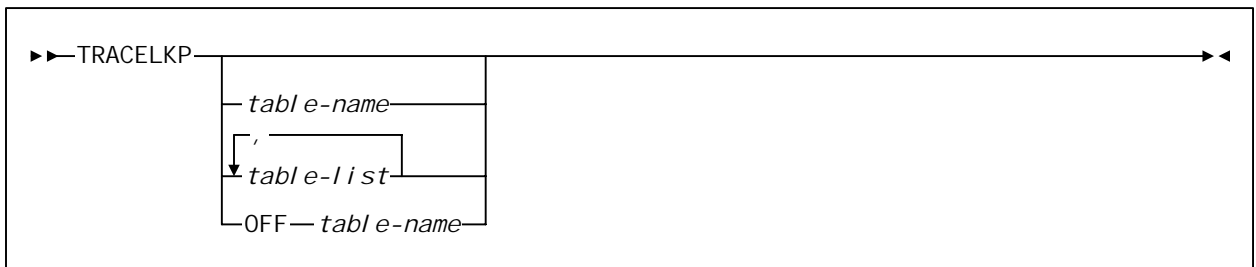
SWitch command



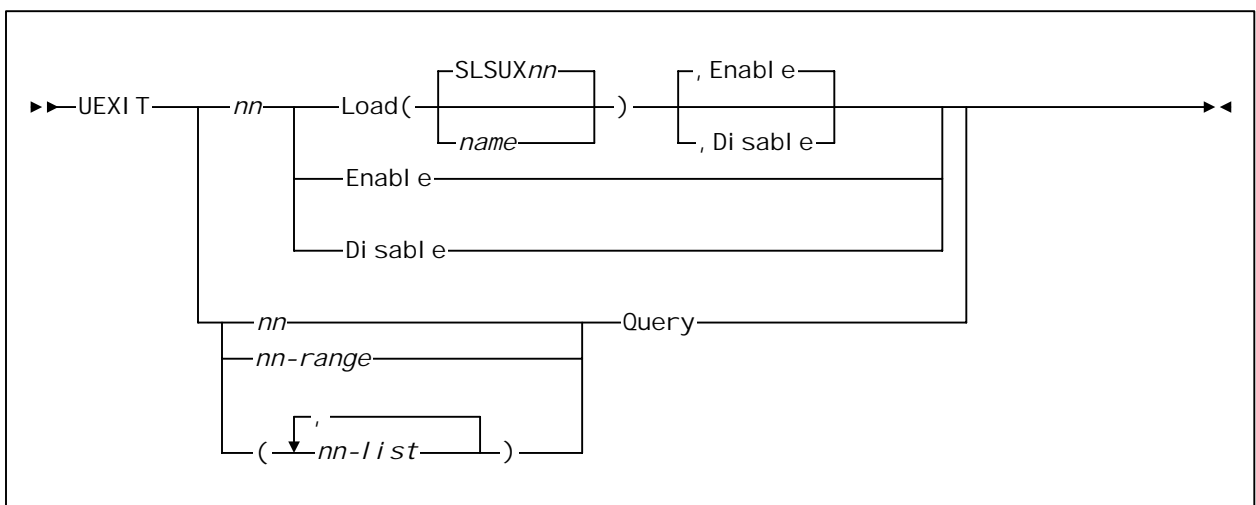
TRace command



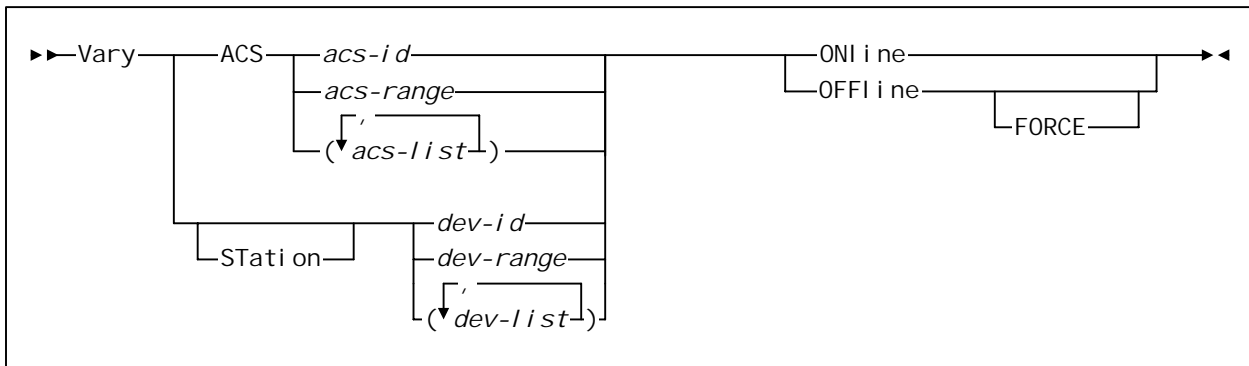
TRACELKP command



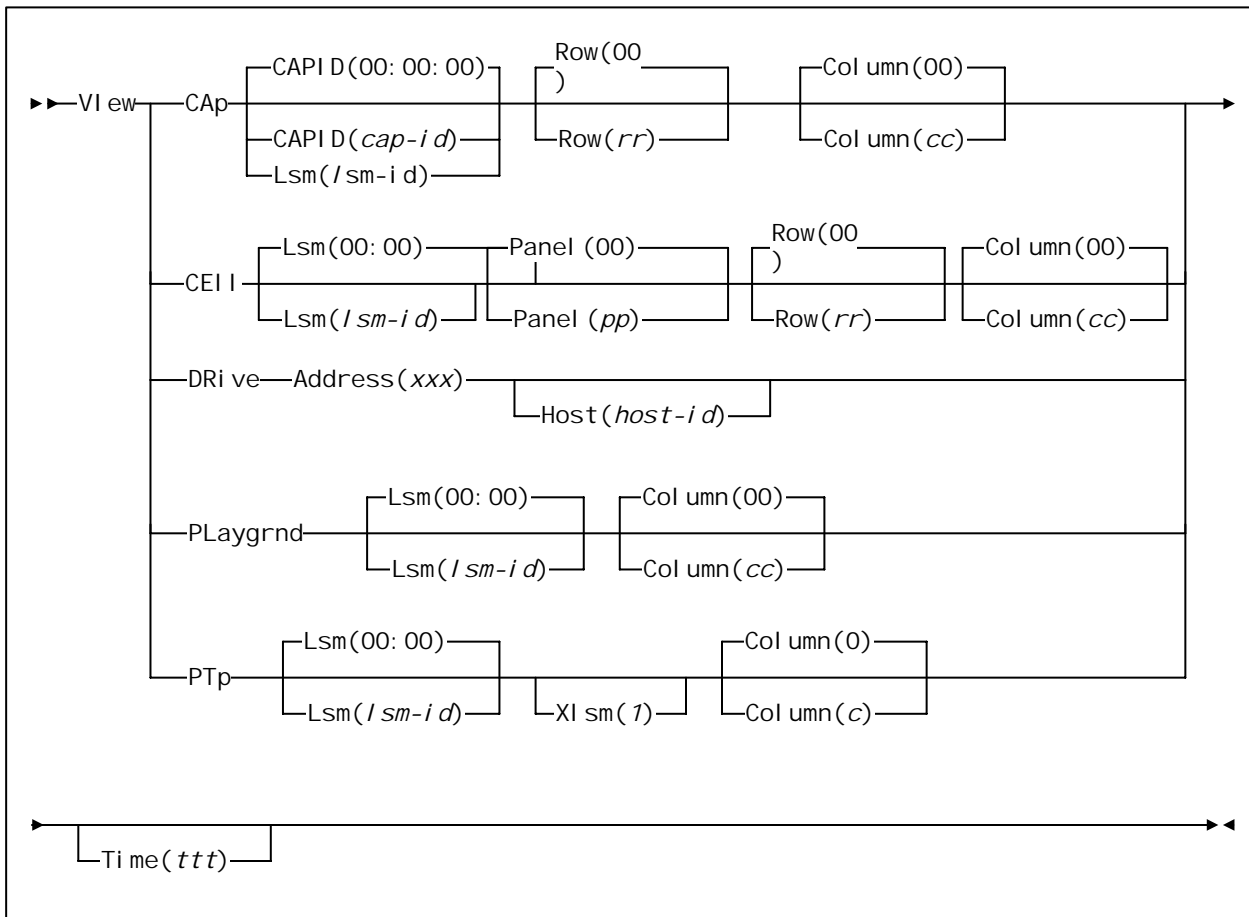
User Exit (UEXIT) command and control statement



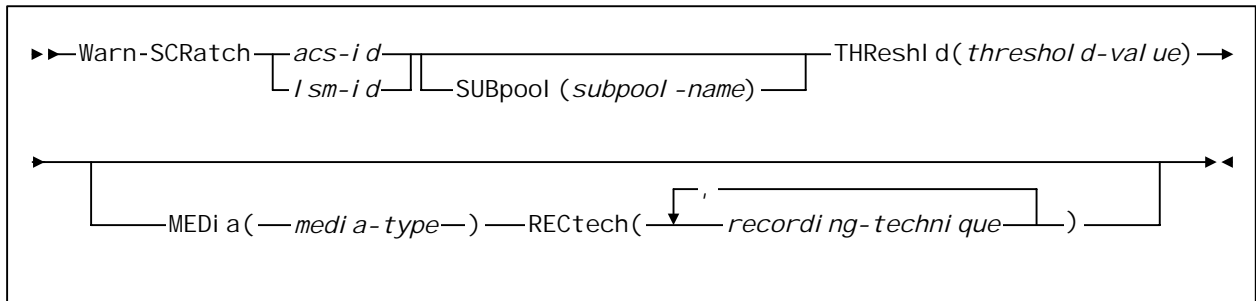
Vary Station command



Vlew command



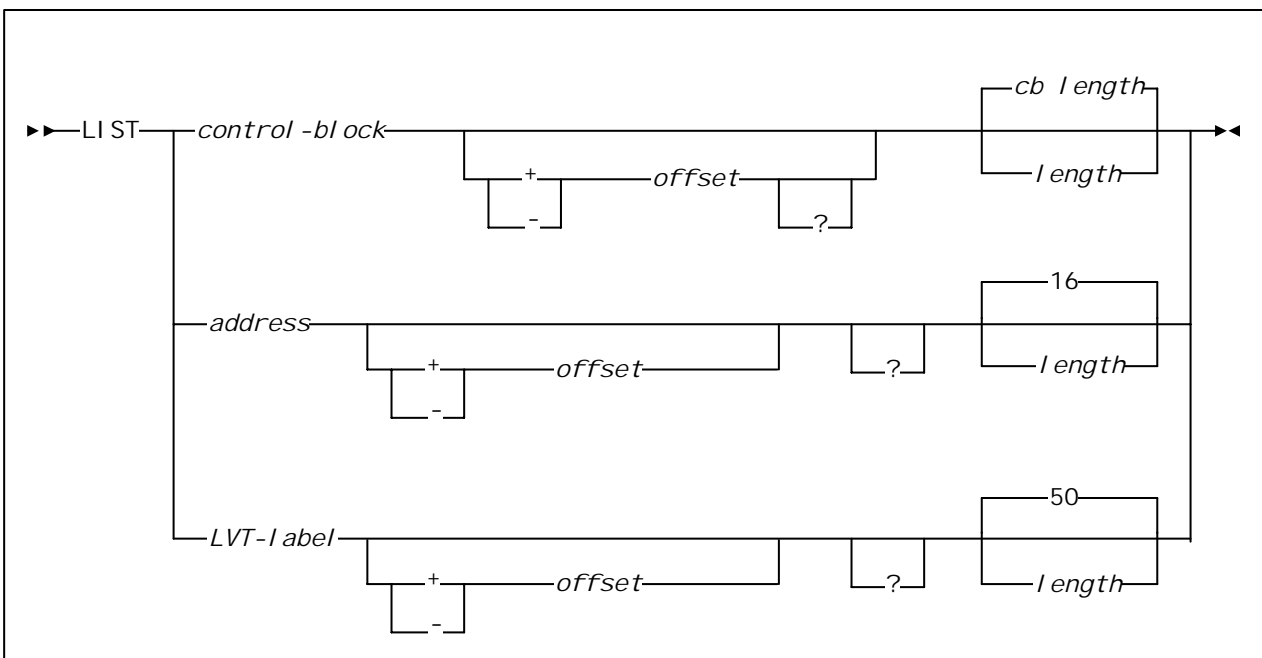
Warn command



HSC Diagnostic Command Syntax

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

List command



TRace command

