

Directory Maintenance VM/ESA



Command Reference

Release 5.0

Directory Maintenance VM/ESA



Command Reference

Release 5.0

Note:

Before using this information and the product it supports, read the general information under "Notices" on page 315.

I Eighth Edition (February 2001)

This edition applies to Version 1, Release 5, Modification 0 of IBM® Directory Maintenance (DirMaint VM/ESA®) (product number 5748-XE4) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC20-1839-06.

© **Copyright International Business Machines Corporation 1979, 2001. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

| | |
|--|-----|
| Preface | vii |
| Who Should Use This Book | vii |
| Where To Find More Information | vii |
| How to Send Your Comments to IBM | vii |
| | |
| Chapter 1. Introduction to DirMaint Release 5.0 Command Support | 1 |
| Release 4.0 Command Compatibility | 1 |
| Using the DirMaint HELP Facility | 2 |
| How to Read the Syntax Diagrams | 3 |
| | |
| Chapter 2. Issuing Commands to DirMaint | 7 |
| Using the Prefix Keywords | 7 |
| DIRMAINT EXEC | 8 |
| | |
| Chapter 3. The DIRMAINT Operands | 15 |
| A Few Words About Defaults | 15 |
| ACCOUNT | 16 |
| ACIGROUP | 19 |
| ACNTADD | 20 |
| ACNTDEL | 22 |
| ADD | 23 |
| AMDISK | 27 |
| APPCPASS | 34 |
| AUTHBY | 37 |
| AUTHFOR | 38 |
| AUTHLINK | 39 |
| AUTHSCIF | 41 |
| AUTOLOG | 43 |
| BACKUP | 45 |
| BATCH | 47 |
| CHECK | 49 |
| CHKSUM | 51 |
| CHNGID | 53 |
| CHVADDR | 56 |
| CLASS | 57 |
| CLEANED | 58 |
| CMDISK | 59 |
| CMS | 66 |
| CONSOLE | 67 |
| COPIED | 68 |
| CP | 69 |
| CPU | 70 |
| CRYPTO | 72 |
| DASDOPT | 74 |
| DATAMOVE | 76 |
| DATEFORM | 78 |
| DEDICATE | 79 |
| DEFAULTS, GLOBALV | 81 |
| DEFINESTAG | 85 |
| DIRECT | 87 |

| | |
|----------------------|-----|
| DIRECTORY | 88 |
| DIREEDIT | 90 |
| DIRMAP | 91 |
| DISABLE | 93 |
| DISTRIB | 94 |
| DLINK | 95 |
| DMDISK | 96 |
| DROPBY | 98 |
| DROPFOR | 99 |
| DROPSCIF | 100 |
| DSECUSER | 101 |
| DUMP | 102 |
| D8ONECMD | 103 |
| ELINK | 104 |
| ENABLE | 106 |
| EXECDROP | 107 |
| EXECLOAD | 108 |
| EXTNCHK | 109 |
| FILE | 110 |
| FREEXT | 111 |
| GET | 113 |
| GETCONSOLE | 115 |
| GLOBALOPTS | 116 |
| GLOBALV, DEFAULTS | 117 |
| HELP | 121 |
| HISTORY | 123 |
| INCLUDE | 124 |
| INVEN | 125 |
| IPL | 126 |
| IUCV | 128 |
| LINK | 131 |
| LOCK | 134 |
| LOGMSG | 135 |
| LOGONBY | 136 |
| MACHINE | 137 |
| MAIL | 138 |
| MAXSPOOL | 139 |
| MAXSTORE, MAXSTORAGE | 140 |
| MDAUDIT | 142 |
| MDISK | 146 |
| MDPW | 149 |
| MINIOPT | 150 |
| MMDISK | 152 |
| NAMESAVE | 154 |
| NEEDPASS | 156 |
| NEWS | 157 |
| NOPDATA | 158 |
| NOTAPE | 159 |
| OFFLINE | 160 |
| ONLINE | 161 |
| OPTION | 162 |
| POOL | 168 |
| POSIXFSROOT | 170 |
| POSIXGLIST | 171 |

| | |
|-------------|-----|
| POSIXGROUP | 172 |
| POSIXINFO | 173 |
| POSIXIUPGM | 175 |
| POSIXIWDIR | 176 |
| POSIXOPT | 177 |
| PRIORITY | 179 |
| PRIOSET | 180 |
| PRIVCLASS | 181 |
| PURGE | 183 |
| PW | 185 |
| PW? | 187 |
| PWGEN | 189 |
| PWMON | 191 |
| QLOG | 193 |
| QUERY, QRY | 194 |
| REPLACE | 195 |
| REVIEW | 197 |
| RLDCODE | 198 |
| RLDDATA | 199 |
| RLDEXTN | 200 |
| RMDISK | 202 |
| SATELLITE | 206 |
| SCAN | 207 |
| SCREEN | 212 |
| SECUSER | 214 |
| SEND | 216 |
| SETACNT | 217 |
| SETCLASS | 219 |
| SETCPU | 220 |
| SETMACH | 222 |
| SETOPTN | 224 |
| SETPRIORITY | 230 |
| SETPW | 231 |
| SETSTAG | 232 |
| SHARE | 234 |
| SHUTDOWN | 236 |
| SPECIAL | 237 |
| SPOOL | 239 |
| STAG | 241 |
| STATUS | 243 |
| STDEVOPT | 245 |
| STORAGE | 246 |
| SUPLIST | 248 |
| SYSAFFIN | 250 |
| TERM | 252 |
| TESTPW | 254 |
| TMDISK | 256 |
| UNLOCK | 258 |
| USEDEXT | 259 |
| USER | 261 |
| USERMAP | 263 |
| USEROPTN | 264 |
| VERIFIED | 268 |
| WORKUNIT | 269 |

| | |
|---|------------|
| XAUTOLOG | 271 |
| XCONFIG | 273 |
| XSTORE | 275 |
| Appendix A. DirMaint Command Sets | 277 |
| Appendix B. Device Characteristics | 283 |
| Maximum Minidisk Sizes and Blocks Per Cylinder for CKD and FBA | 283 |
| Appendix C. The Synchronous Application Programming Interface (SAPI) | 287 |
| Sample Programs | 287 |
| Appendix D. DirMaint Command Summary | 291 |
| Notices | 315 |
| Programming Interface Information | 316 |
| Trademarks | 316 |
| Glossary | 317 |
| Bibliography | 319 |
| DirMaint Library | 319 |
| z/VM Version 3 Release 1.0 Library | 319 |
| VM/ESA Version 2 Release 4.0 Library | 319 |
| Other Related Books | 320 |
| CD-ROM | 320 |
| Index | 321 |

Preface

This is a reference book. The DIRMAINT command operands are listed alphabetically so you can look up their syntax, descriptions, options, and usage notes. The DIRMAINT command and the prefix command descriptions appear before the DIRMAINT command operands.

This book should be used in conjunction with *z/VM: Planning and Administration*.

Who Should Use This Book

This book is meant for those persons responsible for creating and maintaining the VM directory on a VM/ESA system. Some functions in this book are also available to general users, allowing them to implement limited changes to their own directory entries.

Where To Find More Information

For a list of the books that can provide you with additional information on DirMaint, VM/ESA and z/VM, see “Bibliography” on page 319.

How to Send Your Comments to IBM

Your feedback is important in helping us to provide the most accurate and high-quality information. If you have comments about this book or any other VM documentation, send your comments to us using one of the following methods. Be sure to include the name of the book, the form number (including the suffix), and the page, section title, or topic you are commenting on.

- Visit the DirMaint web site at:

<http://www.vm.ibm.com/related/dirmaint>

There you will find the feedback page where you can enter and submit your comments.

- Send your comments by electronic mail to one of the following addresses:

Internet: pubrcf@vnet.ibm.com

IBMLink™: GDLVME(PUBRCF)

- Fill out the Readers' Comments form at the back of this book and return it using one of the following methods:
 - Mail it to the address printed on the form (no postage required in the USA).
 - Fax it to 1-607-752-2327.
 - Give it to an IBM representative.

Chapter 1. Introduction to DirMaint Release 5.0 Command Support

DirMaint Release 5.0 provides support for all VM/ESA Version 2 Release 1.0 directory statements. Most of the DirMaint directory commands have the same names and format as the z/VM directory statements they support. DirMaint also provides additional utilities to help manage minidisk assignments and allocations, and provide a level of security regarding command authorizations and password monitoring.

When DirMaint is first installed, certain default command authorizations exist. The initial command authorizations are shown in Appendix A, “DirMaint Command Sets” on page 277. However, these authorizations are completely tailorable by the installation and therefore are not necessarily the authorizations that currently exist. To find what your current DirMaint authorizations are, consult with your DirMaint administrator.

DirMaint VM/ESA Release 5.0 supports directory statements for various VM environments. Some of the directory statements are supported in All environments, others are only supported in VM/ESA Version 2 Release 1.0 and later. DirMaint Release 5.0 can run in 370, XA, ESA, or XC mode, and is supported on with VM releases:

- VM/ESA (370 Feature) Release 1.5
- VM/ESA Version 1 Release 2.1
- VM/ESA Version 1 Release 2.2
- VM/ESA Version 2 Release 1.0
- VM/ESA Version 2 Release 2.0
- VM/ESA Version 2 Release 3.0
- VM/ESA Version 2 Release 4.0
- z/VM Version 3 Release 1.0

as well as any future releases of VM, until otherwise announced. The release and system level that each DirMaint command is valid in is identified with the **VM Systems:** heading following the syntax diagram. There are some commands that are valid for all environments, but have some operands that are valid only for particular environments. These are noted in the descriptions of the operands.

Release 4.0 Command Compatibility

DirMaint Release 5.0 can run in either of two modes: 140A or 150A. When DirMaint Release 5.0 is running in 140A mode, it accepts the DirMaint commands specified in the release 4.0 syntax. All of the release 4.0 commands are supported, but none of the new release 5.0 function is available. Some DirMaint Release 4.0 commands are obsolete for DirMaint Release 5.0, but are tolerated for compatibility with release 4.0. In time, it is expected the installation will update their user-written routines to take advantage of function provided in 150A mode.

If your installation is running in 140A mode, not all commands in this reference are available. A list of the commands supported in each of these two modes is shown in Appendix A, “DirMaint Command Sets” on page 277.

The differences in how certain commands behave in 140A mode, 150A mode, and Release 4.0 are as follows:

- The new function added to DirMaint's DASD Management support for DirMaint Release 5.0 has made the CLEANED, COPIED, and VERIFIED commands obsolete. These commands no longer serve a useful function in either command level 140A or 150A.
- The changes to DirMaint's DASD Management support for Release 5.0 have made the Release 4.0 syntax of the QRY or QUERY command obsolete. This command no longer serves any useful function in command level 140A. In command level 150A, the syntax is significantly different than the Release 4.0 syntax.
- The command level 140A syntax for the ASSIGN, AUTH, DROP, and UNAUTH commands is compatible with the Release 4.0 syntax. In command level 150A, the ASSIGN and AUTH commands have been combined into a single, enhanced, AUTHFOR command; while the DROP and UNAUTH commands have been combined into a single, enhanced, DROPFOR command.
- The command level 140A syntax and function of the CONSOLE command is compatible with Release 4.0. In command level 150A the same function is performed by the GETCONSOLE command; while the new CONSOLE command allows users to manipulate their CONSOLE directory statement.
- The command level 140A syntax and function of the SETOPTN command is compatible with Release 4.0. In command level 150A the same function is performed by the USEROPTN command; while the new SETOPTN command allows an administrator to set values on the user's OPTION directory statement, without using DIRM GET and DIRM REPLACE.
- Otherwise, the function and syntax of command level 140A commands is compatible with Release 4.0, and the function of command level 150A commands is compatible with both R4 and command level 140A, however the syntax is slightly different.
- Regardless of command level, DirMaint Release 5.0 messages and return codes are different than those in Release 4.0, and there are some differences in the way the commands are processed.

Using the DirMaint HELP Facility

You can receive online information about the DirMaint operands described in this book using the DirMaint HELP Facility. For example, to display a menu of DirMaint HELP information, type:

```
DIRM HELP
```

Place the cursor under a command or topic you want information about and press enter.

To display information about a specific DirMaint operand (ADD in this example), type:

```
DIRM HELP ADD
```

or

```
DIRM HELP AD
```

or

DIRM HELP A

The DirMaint HELP facility recognizes the minimum abbreviation for a DIRMAINT operand.

To display information about the DIRMAINT command, type:

DIRM HELP DIRM

You can also display information about a message (DVH1093 in this example), by typing:

DIRM HELP DVH1093

The DIRMAINT HELP command and its operands are described in "HELP" on page 121.

How to Read the Syntax Diagrams

Throughout this document, command syntax is described using the structure defined below.

- **Flow:**

Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following are symbols used in the syntax diagrams:

- ▶— Indicates the beginning of a syntax diagram.
- ▶ Indicates that the syntax diagram continues on the next line.
- ▶— Indicates that a syntax diagram continues from the previous line.
- ▶◀ Indicates the end of a syntax diagram.

- **Abbreviations:**

Uppercase letters denote the shortest acceptable abbreviation. If an item appears entirely in uppercase letters, it cannot be abbreviated. You can type the item in uppercase letters, lowercase letters, or any combination.

In this example, you can enter KEYWO, KEYWOR, or KEYWORD in any combination of uppercase and lowercase letters.

▶—KEYWOrd—▶◀

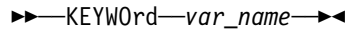
- **Symbols:**

You must code the following symbols exactly as they appear in the syntax diagram.

- * Asterisk
- :
- ,
- = Equal Sign
- Hyphen
- () Parentheses
- .

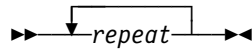
- **Variables:**

Highlighted lowercase items (*like this*) denote variables. In this example, *var_name* represents a variable you must specify when you code the KEYWORD command.

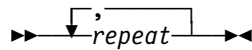


- **Repetition:**

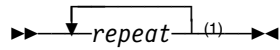
An arrow returning to the left means that the item can be repeated.



A character within the arrow means you must separate repeated items with that character.



A footnote (1) by the arrow references a limit that tells how many times the item can be repeated.

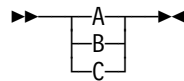


Note:

¹ Specify *repeat* up to 5 times.

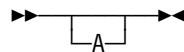
- **Required Choices:**

When two or more items are in a stack and one of them is on the line, you *must* specify one item. In this example, you must choose A, B, or C.

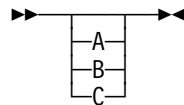


- **Optional Choice:**

When an item is below the line, the item is optional. In this example, you can choose A or nothing at all.

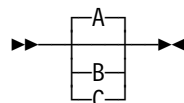


When two or more items are in a stack below the line, all of them are optional. In this example, you can choose A, B, C, or nothing at all.



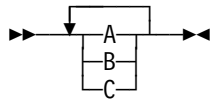
- **Defaults:**

Defaults are above the line. The system uses the default unless you override it. You can override the default by coding an option from the stack below the line. In this example, A is the default. You can override A by choosing B or C.



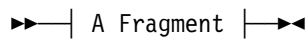
- **Repeatable Choices:**

A stack of items followed by an arrow returning to the left means that you can select more than one item or, in some cases, repeat a single item. In this example, you can choose any combination of A, B, or C.

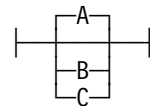


- **Syntax Fragments:**

Some diagrams, because of their length, must fragment the syntax. The fragment name appears between vertical bars in the diagram. The expanded fragment appears in the diagram after a heading with the same fragment name. In this example, the fragment is named “A Fragment.”



A Fragment:



Chapter 2. Issuing Commands to DirMaint

The DIRMAINT command provides the interface between the command issuer and the DIRMAINT service machine. The issuer types the “DIRMAINT” command followed by a command operand. The DIRMAINT command routes the command to the DIRMAINT service machine where the service machine does validation checking and either processes the request or rejects it with an appropriate message. The request sent to the service machine may include prefix keywords. For example:

```
EXEC DIRMAINT REVIEW
```

or

```
DIRM TONODE CHQ1 AUTHBY ADMIN1
```

Using the Prefix Keywords

Additional instructions can be given to the DIRMAINT service machine regarding the processing of a command operand by using a prefix keyword. The prefix keyword (and any operands required for the keyword) is placed before the operand whose processing it is to affect. For example:

```
DIRM AT SYSTEMA MACHINE 370
```

Multiple prefix keywords can be used in a single command string, but each prefix keyword can only be specified once. For example:

```
DIRM TO CHQ1 AS MAINT BY ADMIN1 FOR CEO AT CHQ2 IMMED IPL CMS2
```

Because some prefix keywords allow DirMaint commands to be issued on behalf of another user ID's authority, users must be authorized to use these prefix keywords. Not all DIRMAINT commands accept prefix keywords. Those that do are noted in the “Usage Notes” section of the command description. Prefix keywords and their usage are described under the description of the DIRMAINT command.

Online HELP for Prefix Keywords

Online HELP is available for each individual prefix keyword. To get HELP on a prefix keyword, just type:

```
DIRM HELP prefix_kwd
```

For example, to get online HELP on the FORuser keyword, type:

```
DIRM HELP FOR
```

```
DIRM HELP FORU
```

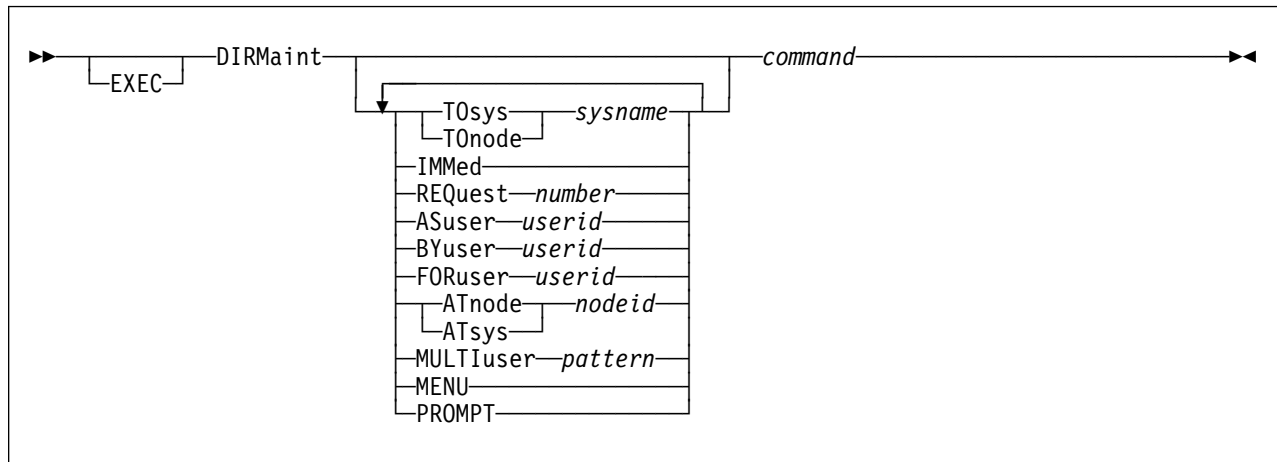
```
DIRM HELP FORUS
```

```
DIRM HELP FORUSE
```

```
DIRM HELP FORUSER
```

DirMaint HELP will accept the minimum valid abbreviation of the keyword.

DIRMAINT EXEC



VM Systems: All

Purpose

The DIRMAINT EXEC provides the command interface between the user and the DIRMAINT virtual machine. The DIRMAINT EXEC sends DirMaint commands to the DIRMAINT service virtual machine for execution.

Operands

EXEC

is optional when the DIRMAINT EXEC is invoked from the CMS command line at the “Ready;” prompt, or when issued within a REXX exec with ADDRESS CMS specified. It is required when the DIRMAINT EXEC is invoked from certain subsystems, or when issued within a non-REXX exec, or within a REXX exec with ADDRESS COMMAND specified.

DIRMaint

is the name of the exec. It may be abbreviated to DIRM unless it is prefaced with the EXEC keyword, or the issuing user is running with SET ABBREVI OFF or SYNONYM (NOSTD).

TOsys *sysname*

TOnode *sysname*

specifies the network node name of the remote system to which the command will be sent for execution.

If the TO parameter is not specified, the default may have been previously set using the DIRMAINT DEFAULTS operand. If the value of the TO parameter is an asterisk (*), either by default or explicit request, the command will be processed within the local system cluster.

The following entries must be defined in the CONFIG* DATADVH file to identify the remote system that TOsys (or TOnode) is to send the DIRMAINT request, unless the IDENTIFY command correctly identifies the network traffic machine, and the user ID of the DIRMAINT service machine on the remote system is DIRMAINT:

FROM= *fromnode* TO= *tonick* S= *spoolid*
 T= *destnode2* U= *serverid*

Where:

fromnode

identifies the network node ID where the transaction originates.

tonick

identifies the nickname for the remote system where the transaction is being sent.

spoolid

identifies the user ID of the machine where punch output should be sent to reach the specified destination. For use within the local system cluster with cross system spooling enabled, this is the user ID of the DIRMAINT service machine; otherwise it is the user ID of an RSCS network service machine.

destnode

identifies the remote system where the transaction is to be processed, as that system is known to the network.

serverid

identifies the user ID of the DIRMAINT service machine on the remote system.

If no entry is found for the specified *sysname* as either a TO= *nickname* or as a T= *destnode*, then a second lookup is attempted using a node ID of asterisk (*). If the *spoolid* is an *, the network service machine's user ID is obtained using the CMS IDENTIFY command. If the *destnode* tag field is an *, the specified *sysname* is used. If the *serverid* is an *, the name DIRMAINT is used.

Note: TOSYS and TONODE are synonymous.

IMMed

The IMMED prefix currently has no specific use in DirMaint Release 5.0. It is accepted to maintain compatibility with previous releases.

REQuest number

associates a numeric value in the range of 1 to 9999 with a specific DirMaint transaction.

This prefix operand is intended to be used primarily with the SAPI interface.

ASuser userid

specifies the user ID under whose privileges a command will be performed. This keyword is primarily intended for use with the TOsys keyword. Unless the BY keyword is also used, the password is verified for the user ID issuing the command. Use of the AS parameter causes a password to be required for authentication, even if you have NEEDPASS NO and even if the ASuser user ID has specified NEEDPASS NO on the target system.

If the AS parameter is not specified, the default may have been previously set using the DIRMAINT DEFAULTS operand. If the value of the AS and BY parameters are both an asterisk (*), either by default or explicit request, any password verification will be done with the user ID from which the command was issued.

The user ID specified with the AS keyword must exist on the target system where the command is to be executed.

If the value of the FOR keyword is an asterisk (*) after default resolution, and the value of the AS keyword is not an *, then the value of the AS parameter will also be used as the value of the FOR keyword.

BYuser *userid*

specifies the user ID that is issuing the command. This keyword is required when the user ID issuing the command is a shared virtual machine that has been logged onto by one of the sharing users with their personal password, and that user does not know the current logon password for the shared virtual machine. Unless the AS keyword is specified, the command is executed with the privileges of the command issuer. The BYUSER keyword is unnecessary if the user issuing the command has set NEEDPASS NO, or if the user issuing the command knows the logon password of the user ID where the command is issued. Use of the BY parameter causes a password to be required for authentication, even if you have NEEDPASS NO and even if the BYuser user ID has specified NEEDPASS NO on the target system.

If the BY parameter is not specified, the default may have been previously set using the DIRMAINT GLOBALV operand. If the value of the AS and BY parameters are both an asterisk (*), either by default or explicit request, any password verification will be done with the user ID from which the command was issued.

The issuing user ID (or the AS user ID) and the user ID specified with the BY keyword must both exist on the target system where the command is to be executed. The issuing user ID (or the AS user ID) must have previously issued an AUTHBY command for the BY user ID. The password must be the correct logon password for the BY user ID on the target system.

FORuser *userid*

specifies the user ID whose directory entry is to be changed. This keyword is optional for all of the general user commands, but is required for most of the privileged user commands.

If the FOR parameter is not specified, the default may have been previously set using the DIRMAINT GLOBALV operand. If the value of the FOR parameter is still an asterisk (*) after default resolution, and the value of the AS keyword is not an *, then the value of the AS parameter is also used as the value of the FOR keyword. If the value of the FOR keyword is still an *, either by default or explicit request, any directory change affects the user ID that issued the command.

When issued by a general user, the user ID specified with the FOR keyword must have previously issued an AUTHFOR command to authorize the issuer (or the AS user ID).

ATnode *nodeid***ATsys** *nodeid*

specifies the system node name within a multiple system cluster that the directory change is to affect, without affecting other nodes in the cluster. If the directory statement to be affected already exists in the directory with internal SYSAFFIN statements that conflict with this request, the command will be rejected by the DIRMAINT service machine.

If the AT keyword is not specified, the default may have been previously set using the DIRMAINT GLOBALV operand. If the value of the AT keyword is an asterisk (*), either by default or explicit request, any directory change will affect all nodes within the cluster on which the command is processed.

Note: ATNODE and ATSYS are synonymous.

MULTIuser

identifies one or more users (through the use of a scan pattern) to be acted upon by the specified command. The pattern supplied with the MULTIUSER prefix operand is used to build a list of users. Once the list has been built, a batch file is generated to do the actual commands. Each batch command is issued using the invoker's authority. The pattern uses a "wild card" character in much the same way that the CMS LISTFILE command uses it. The following rules apply:

- A single asterisk (*) matches all users on the system.
- Multiple, adjacent asterisks are considered to be a single *, (for instance, ABC***123 is equal to ABC*123).
- Entries \$DIRCTL\$, \$DIRGRP\$, ALL, and SYSTEM are always excluded from matches and are never the target of a MULTIUSER command.
- Leading, trailing, and imbedded asterisks may be used in a pattern.
- Multiple asterisks may be used within a single pattern (for instance, H*WL*ND* or *O*A*).

Users of this prefix operand must be authorized by the MULTIUSER_VERIFICATION_EXIT. This exit controls what users are authorized to use the MULTIUSER prefix operand. In addition, this prefix operand can not be used with some DIRMAINT commands (the specific commands are listed in the sample exit routine). This exit routine must be altered by the local installation to enable this support. As shipped, the exit routine rejects all transactions.

The MULTIUSER_VERIFICATION_EXIT also has the ability to provide a list of users (overriding the pattern matching) for use with the provided command. This allows the local site to use whatever method is appropriate to subset the users contained in the USER DIRECT file. Details on the steps required to utilize this capability are documented in the sample exit routine.

If this prefix is used in conjunction with the FORUSER prefix operand, the FORUSER operand is ignored.

MENU

specifies that a menu be displayed (if available) for the specified command keyword prior to its execution.

Any parameters provided beyond the command keyword will be ignored by the menu processor. The command string will be built by the menu processor and passed to DirMaint for evaluation and execution.

The MENU prefix operand is implied if the command string given to DirMaint requires parameters beyond the command key word but they are not provided.

DIRMAINT

Other prefix operands can be specified in conjunction with the MENU prefix operand and are kept/held/used by the command built by the menu processor. For example:

```
DIRM FOR USERAA MENU PW
```

If a menu is not available for the command keyword, the command string will be passed to DirMaint for evaluation and execution as entered. Menus are not provided for the following DirMaint commands:

- ?
- CLEANED
- COPIED
- VERIFIED

PROMPT

requests that password prompting be done if passwords are not provided on the command line for specific operands of the DIRMAINT command.

This prefix operand is accepted on all commands but is primarily intended for use with the APPCPASS, LINK, and MDISK operands of the DIRMAINT command.

command

any DirMaint command.

Usage Notes

1. One or more of the prefix keywords (TOsys, IMMED, REQuest, ASuser, BYuser, FORuser, ATnode, MENU, PROMPT) may be specified in any order, but each keyword may only be specified once.
2. Most DirMaint commands are sent to the DIRMAINT virtual machine for processing. You will usually be prompted to enter a logon password for authentication unless you have issued a DIRM NEEDPASS NO command. If the AS or BY keyword parameters are specified, you will be prompted to enter a logon password for authentication regardless of the value of the NEEDPASS option. If you are running as a disconnected service machine, you may stack the password before invoking the DIRMAINT EXEC.
3. There are a few DirMaint commands that are completely processed within the virtual machine of the command issuer. No transaction is sent to the DIRMAINT service machine. Therefore, no password is required or requested for authentication, even if you have set NEEDPASS to YES. If you are running as a disconnected service machine, nothing will be pulled from the stack. The following commands are completely processed within the issuer's virtual machine:
 - CHECK
 - DEFAULTS
 - EXECDROP
 - EXECLOAD
 - GLOBALV
 - HELP
 - any command with an invalid syntax
4. The DIRMAINT command itself is not valid within a BATCH command file. The TOsys, ASuser, and BYuser prefix operands are not valid within a BATCH command file. However, they may be used as a prefix to a

BATCH command. The FORuser and ATnode prefix commands are valid within a BATCH command file, and they may also be used as a prefix to a BATCH command. When used as a prefix command to a BATCH command, the FORuser and ATnode operands set a default that is used for all commands within the BATCH file, unless overridden by another FORuser or ATnode prefix for individual commands within the BATCH file.

5. If you are planning to issue DIRMAINT commands from exec programs and perform actions based upon message responses from the DIRMAINT service machine, refer to Appendix C, "The Synchronous Application Programming Interface (SAPI)" on page 287 for instructions on how to set up your execs to activate the Synchronous Application Programming Interface (SAPI) and access the response information.
6. If the PRESET global variable, in the DVH15 pool, has a value other than blanks or NOLOG, DirMaint will send that value to the DIRMAINT server without prompting for a password. To prevent disclosure of that password, the variable will be reset to a value of NOLOG upon exit, thus the value must be set prior to each call to the DIRMAINT EXEC.
7. The NEEDPASS command is an exception to the PREFIX KEYWORDS, ignoring any PREFIX keyword that may have been supplied on the command.

Examples

EXEC

```
EXEC DIRMAINT REVIEW
```

TOsys, TOnode

```
DIRMAINT TO SYSTEM1 ENABLE
```

```
DIRM GLOBALV DTO *
```

```
DIRM GLOBALV DTOSYS SCHOOL
```

```
DIRM TONODE CHQ1 AUTHBY ADMIN1
```

```
DIRMAINT TOSYS CHQ1 BY ADMIN1 FOR CEO AT  
CHQ2 IMMED IPL CMS2
```

```
password
```

```
DIRM TO SCHOOL FOR * BY * AUTHBY PROF2
```

IMMED

```
DIRM IMMED AT SYSTEMA MACHINE 370
```

REQuest

```
DIRM REQ 6 SECUSER EDWARDSR
```

ASuser

```
DIRMAINT ASUSER ALTEREGO ACCOUNT PROJECT1
```

```
password
```

ATnode ATsys

```
DIRM AT SYSTEMA MACHINE 370
```

BYuser

DIRMAINT

DIRMAINT AUTHBY OWNER
DIRM BY OWNER ONLINE
password

DIRM TO CHQ1 AUTHBY ADMIN1
DIRM TO CHQ1 BY ADMIN1 FOR CEO AT
CHQ2 IMMED IPL CMS2
password

FORuser

DIRM AUTHFOR ISSUER
DIRM FOR TARGETID DISTRIB NEWDEPT

DIRM TO CHQ1 AUTHBY ADMIN1
DIRM TO CHQ1 BY ADMIN1 FOR CEO AT
CHQ2 IMMED IPL CMS2
password

XEDIT SETPW CLASS101 A
INPUT FOR STUDENT1 SETPW CLASS101
INPUT FOR STUDENT2 SETPW CLASS101
INPUT FOR STUDENT3 SETPW CLASS101
INPUT FOR STUDENT4 SETPW CLASS101
INPUT FOR STUDENT5 SETPW CLASS101
INPUT FOR STUDENT6 SETPW CLASS101
INPUT FOR STUDENT7 SETPW CLASS101
INPUT FOR STUDENT8 SETPW CLASS101
INPUT FOR STUDENT9 SETPW CLASS101
FILE
DIRM BATCH SETPW CLASS101 A

MULTIuser

DIRM MULTIUSER G32* DIST G32/B90
DIRM MULTIUSER *USER* ACCOUNT NEWACNT

MENU

DIRMAINT MENU REVIEW

PROMPT

DIRMAINT PROMPT MDISK 191 MR

Chapter 3. The DIRMAINT Operands

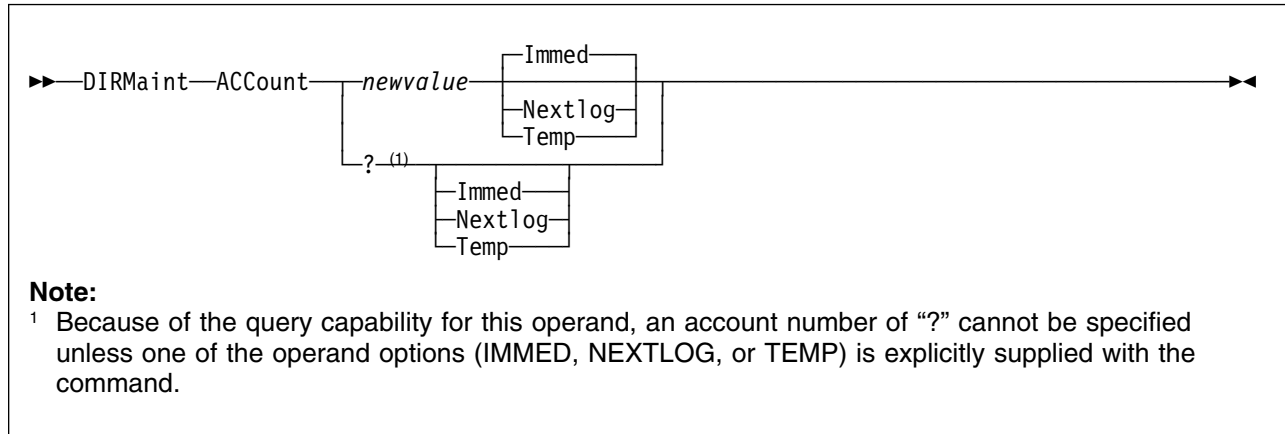
This section describes the syntax, environment, purpose, operand descriptions, and any additional usage information regarding the DIRMAINT command operands. In some cases, examples of usage are provided.

A Few Words About Defaults

For DirMaint directory commands, any defaults that you assume when issuing the command will be the same as the defaults in place for the release of VM/ESA installed on your system. Because of this, the VM/ESA defaults are not shown in the syntax diagrams. However, any defaults set by DirMaint are shown.

Ensure that you have a copy of *z/VM: Planning and Administration* for the appropriate VM/ESA version and release installed on your system to reference the directory statement default settings.

ACCOUNT



VM Systems: All

Purpose

The ACCOUNT operand of the DIRMAINT command enables you to use an account number other than the one defined as “currently in use” in the ACCOUNT control statement. You can determine the “current” account number by issuing:

```
DIRM ACCOUNT ?
```

If your installation has provided alternate account numbers for you to select from, they must be defined in your directory entry. The method of providing alternate account numbers is a function of the operating system environment and your installation's policy. For more information on defining and preparing alternate account numbers for your installation, see *z/VM: Planning and Administration*.

Operands

newvalue

is the new account number to be brought into use for your virtual machine. It must be one of the alternate account numbers defined in your directory entry. To determine the alternate account numbers, issue the DIRM REVIEW command operand to obtain a copy of your directory entry and inspect the appropriate statements. By inspecting the optional *AC= statement and the secondary account numbers on the ACCOUNT statement you can determine what account numbers are valid for you to use. You may select any one of the 1 to 8 character account numbers defined.

Immed

specifies that the new account number is to become effective immediately and remain in effect until changed. The source directory is updated to reflect the change. This is the default value.

Nextlog

specifies that the new account number is to become effective with the next logon and remain in effect until changed. The source directory is updated to reflect the change.

Temp

specifies that the new account number is to be used only for the current terminal session. The source directory is not changed.

? allows you to query your current account number.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For a complete description of the ACCOUNT directory statement, see *z/VM: Planning and Administration*.
3. Use of the IMMED and TEMP options require that the CP update-in-place facility has been enabled by the installation. This is done by placing an UPDATE_IN_PLACE=YES record in the DIRMAINT configuration file. If this is not the case, the user will be informed via message. The transaction will not take place.
4. The directory is altered when using the IMMED and NEXTLOG options. The TEMP option updates the account number for the current session only.
5. The *AC= statement is an optional statement containing one or more valid secondary account numbers. For more information, see "SETACNT" on page 217.
6. When determining the validity of an account number, the directory and its included profile are consulted to build a list of candidate account numbers. The effective ACCOUNT and *AC= statements are used for this purpose. The effective statement is defined as the first statement found by looking at the directory and then the profile. For example, if the statement occurs in both the user directory and the profile, the user directory statement is used.
7. If a locally installed ACCOUNT_NUMBER_VERIFICATION_EXIT returns control to the ACCOUNT command handler with a return code of zero, no additional checking will be done on the account number, and it will be accepted without consulting the secondary account numbers or the *AC= statement.
8. When altering an existing account record, the current primary account number is replaced with the new number, provided the new number passed all required validity checks. The current account number is not placed on the secondary list unless the new primary account number was validated by consulting the secondary list. If the current primary account number is not on the *AC= statement and not on the current list of secondary account numbers, it is destroyed during the replace process and can not be recovered.
9. When altering an account record in a directory where an account record did not exist, the account record in the profile is used (if it exists) as the effective account statement. Any secondary account numbers and the

ACCOUNT

distribution code on the account record from the profile are placed into the user directory entry along with the new primary account number.

For example, if the user directory contains no ACCOUNT statement, but does include the following *AC= statement:

```
*AC= MARK0001 MIKE0002 CHUCK003
```

And, in addition, the user directory includes a profile that contains the following account statement:

```
ACCOUNT DEPTG32 BOX90C PROJ001 PROJ002 PROJ003
```

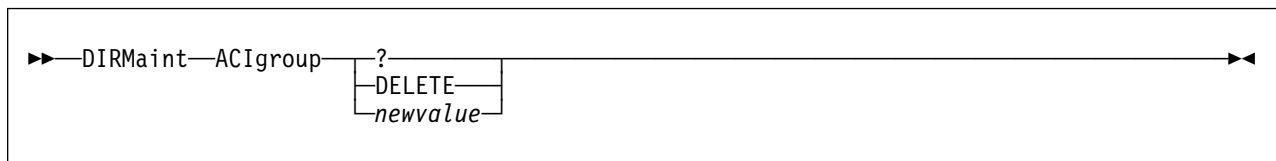
the command:

```
DIRM ACCOUNT MIKE0002
```

will result in the following ACCOUNT statement being placed into the directory entry:

```
ACCOUNT MIKE0002 BOX90C PROJ001 PROJ002 PROJ003
```

ACIGROUP



VM Systems: All

Purpose

The ACIGROUP operand of the DIRMAINT command enables you to specify, change, query or delete the ACIGROUP statement in the current user directory.

The ACIGROUP statement is used to specify the name of the group to which this user is assigned.

Operands

? allows you to query your ACIGROUP setting.

DELETE

allows you to delete your ACIGROUP statement.

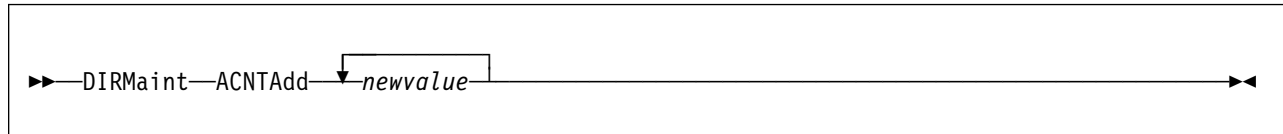
newvalue

is the new or changed group name to be associated with this user ID. If specified it will replace any existing value on an existing ACIGROUP statement.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For a complete description of the ACIGROUP directory statement, see *z/VM: Planning and Administration*.
3. Due to the syntax of this command, ? and DELETE are not allowed as ACI group names. If one of these values is required as an ACI group name, use DIRM GET and DIRM REPLACE to set it.

ACNTADD



VM Systems: All

Purpose

The ACNTADD operand of the DIRMAINT command enables you to add an account number to your list of secondary account numbers on your ACCOUNT control statement. You can determine what your current account control record looks like by issuing:

```
DIRM ACCOUNT ?
```

The method of providing alternate account numbers is a function of operating system environment and installation policy. For more information on defining and preparing alternate account numbers for your installation, see *z/VM: Planning and Administration*.

Operands

newvalue

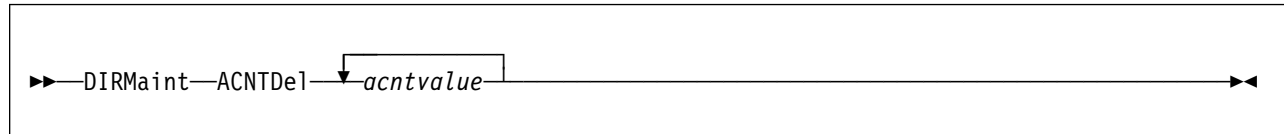
specifies an account number to be placed on the ACCOUNT control record as a secondary account number. 1 to 7 *newvalue* parameters are allowed. *newvalue* must be one of the alternate account numbers defined in your directory entry, or be specifically authorized by a local exit. To determine the alternate account numbers, issue the DIRM REVIEW command operand to obtain a copy of your directory entry and inspect the appropriate statements. By inspecting the *AC= statement you can determine what account numbers are valid for you to use. You may select any one of the 1 to 8 character account numbers defined.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. When determining the validity of an account number, the directory and its included profile are consulted to build a list of candidate account numbers. The effective *AC= statement is used for this purpose. The effective statement is defined as the first statement found by looking at the directory and then the profile. That is, if the statement occurs in the user directory and the profile, the user directory statement is used.
3. If a locally installed ACCOUNT_NUMBER_VERIFICATION_EXIT returns control to the ACNTADD command handler with a return code of 0, no additional checking will be done on the account number and it will be accepted without consulting the *AC= statement.

4. When adding an account number (or numbers) to the secondary list on the account number, the entire request will be rejected if a number on the add list already exists on the current list of secondary account numbers.
5. If you are using this command to add secondary account numbers to a nonexistent account statement in a user directory, the primary account number and distribution code default to the directory name after checking the profile for an existing account statement. If the account statement exists but contains only the primary account number, the distribution code defaults to the directory name.
6. If you attempt to place more than seven account numbers on the secondary list, the entire request is rejected. The amount is determined by adding the existing secondary account number and the proposed new account numbers.
7. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message when the source directory has been updated and when the change has been placed online.

ACNTDEL



VM Systems: All

Purpose

The ACNTDEL operand of the DIRMAINT command enables you to remove an account number (or numbers) from your list of secondary account numbers on your ACCOUNT control statement. You can determine what your current account control record looks like by issuing:

```
DIRM ACCOUNT ?
```

Operands

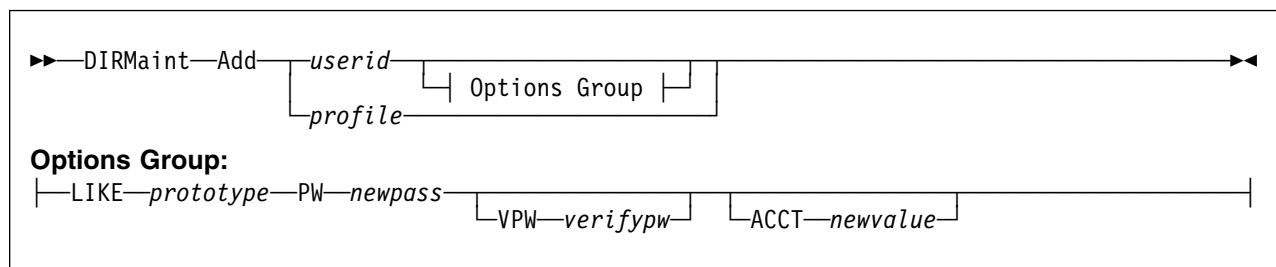
acntvalue

is a list of 1 to 7 account numbers to be removed from the ACCOUNT control record secondary account number list.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The specified account numbers must exist on the current ACCOUNT control statement. If a specified number does not exist on the control statement the entire request will be rejected.
3. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message when the source directory has been updated and when the change has been placed online.

ADD



VM Systems: All

Purpose

The ADD operand of the DIRMAINT command is used to add an entry to the directory for a new user ID or profile. The configuration of the user can be created in the ADD operand invoker's virtual machine and forwarded to DIRMAINT, or the user's configuration can be taken from a prototype definition stored in the DIRMAINT server. PROFILE definitions cannot be added using the LIKE keyword.

Operands

userid

identifies the user ID of the new virtual machine you are creating. The user ID must be the file name of a file with a file type of DIRECT containing the definition for the user ID being created, unless the LIKE parameter specifies the file name of an existing prototype.

LIKE *prototype*

indicates that the new user directory entry is to be added LIKE the specified stored *prototype* file.

PW *password*

specifies the logon password to be installed in the created user definition for the new user.

ACCT *newvalue*

optionally specifies an account number to be installed in the created user definition.

VPW *verifypw*

specifies the same password again for typographical verification.

profile

is the name of the new PROFILE definition you are creating.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOSys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The LIKE keyword can be used to create a user configuration that matches a prototype created by the installation and stored on the DIRMAINT service

machine. The specified prototype may be modeled after an existing user directory entry (that is, *userid* DIRECT).

When using the LIKE keyword, the DIRMAINT server searches its accessed disks for a prototype definition with a file name of *prototype* and a file type of PROTODIR. This prototype directory is then altered with the provided *userid* and *newpass* information before being added to the system.

3. You can easily add profiles or user IDs by keeping skeletal directory definitions in *profile* DIRECT and *userid* DIRECT files containing the standard control statements. These files can be copied and the file name or profile name changed to the new user ID or new profile name you are adding. The new *userid* DIRECT or *profile* DIRECT file can then be modified as necessary with the CMS editor. For example, you might keep files like the following two on your own minidisk and use them as models for adding new profiles and user IDs. To make a new profile or user control statement, you would insert the new profile name in the PROFILE control statement, or the new user ID and CP logon password in the USER control statement. Then modify the other fields as appropriate.

```
PROFILE profile
  CONSOLE 009 3215
  SPOOL 00C 2540 R
  SPOOL 00D 2540 P
  SPOOL 00E 1403
  SPECIAL OFF TIMER
  LINK MAINT 190 190 RR
  LINK MAINT 19E 19E RR
```

```
USER userid password 4M 16M G 64
  INCLUDE PROFILEX
  ACCOUNT 99999999 distcode
MDISK 191 XXXX AUTOG 004 GROUPNAM MR
```

4. The AMDISK and MDISK control statements can specify automatic allocation on a specific *valid* by using the AUTOV keyword instead of a starting cylinder or block number as in this example:

```
MDISK 191 XXXX AUTOV 008 DSKPM8 MR
```

The example specifies an 8 cylinder minidisk to be allocated an extent on volume DSKPM8. The installation EXTENT CONTROL file must contain the required entries.

```
MDISK 191 XXXX AUTOV 008 DSKPM8 MR READ1 WRITE2
```

The example specifies an 8-cylinder minidisk to be allocated an extent on volume DSKPM8. The installation EXTENT CONTROL file must contain the required entries. This example also sets the read and write passwords to READ1 and WRITE2, respectively.

Note: For more examples of DirMaint automatic allocation methods, see “AMDISK” on page 27.

5. The AMDISK and MDISK control statements can also take the traditional form in which you specify all parameters including *devtype*, a specific *volser*, and the starting cylinder or block number.

6. The AMDISK and MDISK control statements are stripped from the new directory entry when ADD processing begins. A batch request is generated for each MDISK statement and ADD processing continues. ADD processing will reorder the contents of user DIRECT file. The comments may no longer be paced as desired. Should the MDISK (AMDISK) request fail, the user will be notified of the error as the request is processed.

Note

The MDISK statements are removed from the new directory entry and passed directly to the DIRMAINT server for AMDISK command processing. A not valid syntax in the new directory entry will result in an error from the AMDISK command processor and the entire ADD request will fail.

The new directory entry you are adding can contain either the MDISK keyword or the AMDISK keyword. If the new entry contains the MDISK keyword, the command is passed to the DIRMAINT server using the AMDISK 140A mode syntax. If the AMDISK keyword is used, the command is passed to the DIRMAINT server using the 150A mode syntax.

Note

The ADD Command issuer will need to have the proper 140A and 150A cmdlevel authority or else the ADD request will fail.

The 150A command syntax provides a richer set of allocation request options than the 140A command syntax. Both have specific syntax requirements on the command that must be followed when entered into the directory entry being added. For instance, the AMDISK syntax for the 150A command set requires the use of the PWS keyword when providing passwords. So

```
AMDISK 191 XXXX AUTOV 008 DSKPM8 MR READ1 WRITE2
```

is syntactically invalid because the required keyword PWS is not provided. The correct syntax would be

```
AMDISK 191 XXXX AUTOV 008 DSKPM8 MR PWS READ1 WRITE2
```

However, if the inbound directory entry contained

```
MDISK 191 XXXX AUTOV 008 DSKPM8 MR READ1 WRITE2
```

DIRMAINT will consider this a 140A command level request and the command will process correctly.

7. LINK statements are handled via batch in the same way that MDISK statements are handled. They are removed from the new directory and added individually.
8. If the new entry (user or profile) contains statements in error, the entire ADD request is rejected.
9. User IDs \$DIRCTL\$, \$DIRGRP\$, SYSTEM, and ALL are restricted. Attempts to add these user IDs are rejected.
10. If an INCLUDE statement in a new user entry invokes a nonexistent profile, an error message is issued and the entry is rejected.

11. If an attempt is made to add a user "LIKE" a stored prototype that does not exist on the DIRMAINT server, an error message is issued and the entry is rejected.
12. As the user statement within a stored prototype directory entry must be modified to reflect the user ID and password of the entry being added, it is possible that the addition of the user ID/password entries to the statement could cause its length to exceed 71 characters. If this occurs the ADD request is rejected.
13. A way to add a user with a pre-defined configuration is to first create a "prototype" file under the name "*fn* PROTODIR." Install it on any disk on the DIRMAINT server using the FILE command. For example:

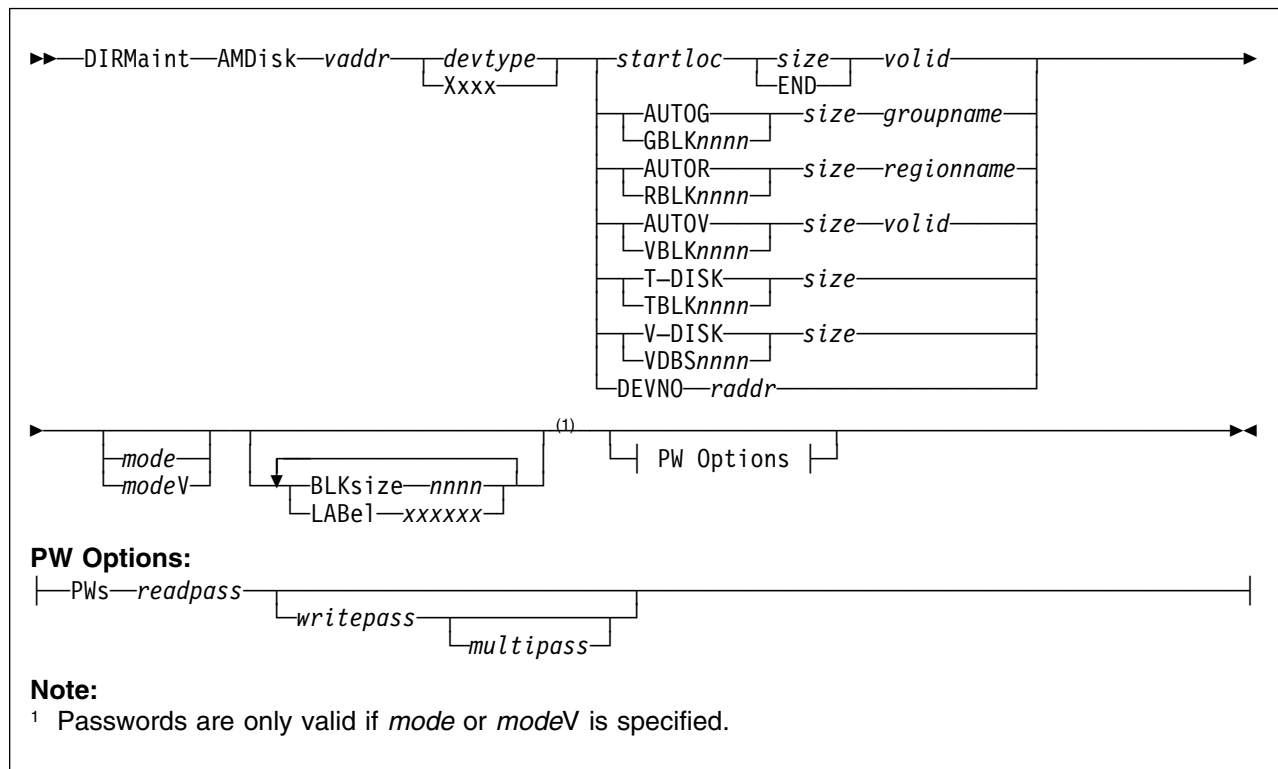
```
DIRM FILE fn PROTODIR A
```

As many unique PROTODIR files as needed can be installed in this manner. The user ID inside such files on the USER card must match the file name under which the prototype file is installed. All minidisks in the prototype definition must be allocated using automatic minidisk allocation. The new password must be specified when adding a user "LIKE" the prototype. An account number is optional.

If the prototype definition includes an ACCOUNT or a *AC= record, each account number is checked by the ACCOUNT_NUMBER_VERIFICATION_EXIT. If the ACCT newvalue option is specified on the command invocation, the newvalue must be contained in the list of valid secondary account numbers on the ACCOUNT statement within the named PROTODIR file, or in the list of valid tertiary account numbers on *AC= records within the named PROTODIR file, or the ACCOUNT_NUMBER_VERIFICATION_EXIT must be enabled to treat the ADD-ACCT command as exempt from this requirement; otherwise, the ADD command will be rejected.

14. An MDISK directory statement may be followed by DASDOPT or MINIOPT directory statements. An AMDISK statement may also be followed by DASDOPT or MINIOPT statements, provided that neither the BLKSIZE nor the LABEL keywords have been used on the AMDISK record. If either of the BLKSIZE or LABEL keywords have been used on the AMDISK record, the minidisk is assigned to the DATAMOVE machine(s) for formatting, and processing for a subsequent DASDOPT or MINIOPT will not find the associated parent MDISK statement in the user's directory entry, causing the ADD to fail, with message DVHMOP3302E.

AMDISK



VM Systems: All

Purpose

Use the AMDISK operand of the DIRMAINT command to add a new minidisk to a user definition.

Operands

vaddr

is the virtual device address of the new minidisk.

devtype

is the device type of the volume to which the minidisk is assigned. Otherwise, if automatic allocation is used, specify the device type as Xxxx. Automatic allocation options include:

- AUTOG and GBLK
- AUTOR and RBLK
- AUTOV and VBLK

operations. The real device type will be taken from the REGIONS section in the EXTENT CONTROL file for that volser.

startloc

is a decimal cylinder relocation factor or a decimal FB-512 device block number that specifies the beginning of the minidisk extent corresponding to cylinder 0 or block 0 of the user's minidisk.

It is not advisable to define a minidisk starting at real cylinder 0, unless it is to be used by operating system (OS) indexed sequential access method (ISAM), in which case it must begin at real cylinder 0. If you do assign a minidisk beginning at real cylinder 0, the user who owns it must realize that the minidisk label is the real label that the user and the system use to identify the disk. Remember that CP-owned volumes must not have minidisks beginning at real cylinder 0. Generally, CKD DASD extents should begin with cylinder 1 or higher, and FBA DASD extents should begin with block 32 or higher.

AUTOG

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a volume within a specified group. The group must have been defined in the EXTENT CONTROL file, and the regions within the group must also be defined.

GBLKnnnn

is functionally equivalent to AUTOG except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: GBLK512, GBLK0512, GBLK1024, GBLK2048, GBLK4096, GBLK1K, GBLK2K, and GBLK4K. In addition, GBLK800 or GBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

The disk will not be formatted for CMS use by DirMaint unless the BLKSIZE or LABEL operands are used, but the number of CMS blocks of a given size per cylinder is used to calculate the number of CKD cylinders or FB-512 blocks necessary to satisfy the request.

AUTOR

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a specified region or volume. The region must have been defined in the EXTENT CONTROL file.

RBLKnnnn

is functionally equivalent to AUTOR except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: RBLK512, RBLK0512, RBLK1024, RBLK2048, RBLK4096, RBLK1K, RBLK2K, and RBLK4K. In addition, RBLK800 or RBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

The disk will not be formatted for CMS use by DirMaint unless the BLKSIZE or LABEL operands are used, but the number of CMS blocks of a given size per cylinder is used to calculate the number of CKD cylinders or FB-512 blocks necessary to satisfy the request.

AUTOV

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a specified volume. The volume must have been defined in the EXTENT CONTROL file.

VBLKnnnn

is functionally equivalent to AUTOV except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: VBLK512, VBLK0512, VBLK1024, VBLK2048, VBLK4096, VBLK1K, VBLK2K, and VBLK4K. In addition,

VBLK800 or VBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

The disk will not be formatted for CMS use by DirMaint unless the BLKSIZE or LABEL operands are used, but the number of CMS blocks of a given size per cylinder is used to calculate the number of CKD cylinders or FB-512 blocks necessary to satisfy the request.

T-DISK

provides temporary disk space to a user upon logging on. This space is made available from preallocated system disk space. The owner must initialize or format this minidisk at each logon. It will be a part of the owner's virtual configuration until logoff or disk detachment, at which time the owner loses all files and the space is returned for reallocation to another user. If concerned about security, the owner should initialize or reformat the minidisk before releasing it.

TBLKnnnn

is functionally equivalent to T-DISK except that the size is specified in units of CMS 512, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: TBLK512, TBLK0512, TBLK1024, TBLK2048, TBLK4096, TBLK1K, TBLK2K, and TBLK4K.

V-DISK

provides virtual disk space to a user upon logging on. The V-DISK may already exist when the user links to it, providing other users that previously linked to it are still linked when the user logs on. Otherwise, the owner must initialize or format this minidisk at each logon. It remains a part of the owner's virtual configuration until logoff or disk detachment. When the last linked user logs off or detaches the V-DISK, the file space is returned for reallocation to another user.

VDBSnnnn

is functionally equivalent to V-DISK except that the size is specified in units of CMS 512, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: VDBS512, VDBS0512, VDBS1024, VDBS2048, VDBS4096, VDBS1K, VDBS2K, and VDBS4K.

size

is a 1 to 8 digit decimal number specifying the size of the new minidisk in CKD cylinders or FBA blocks. For allocation requests involving block sizes, allocation will be converted to the exact number of FB-512 blocks for FBA devices or to the minimum number of cylinders needed to contain the specified number of CMS blocks for CKD devices.

END

specifies that the minidisk will extend from the assigned starting cylinder to the end (last cylinder) of the volume in which it is defined. This parameter is valid for fixed allocation only.

valid

is the volume label of the DASD volume, 1 to 6 alphanumeric characters, on which the minidisk is to be allocated, or the reserved identifier of &SYSRES. This applies to fixed allocation (where a specific starting cylinder or block number is specified) and to automatic allocation on a volume (AUTOV, VBLKnnnn).

groupname

specifies the name of the DASD allocation group from which DirMaint will pick the *volser* and starting CKD cylinder or FBA block. Groups consist of one or more specific *regions*. This is valid for AUTOG and GBLKnnnn requests only. Groups must be defined in the EXTENT CONTROL file, and can be 1 to 8 alphanumeric characters.

regionname

is the name associated with the *region*, 1 to 8 alphanumeric characters, on which the minidisk is to be allocated. This is valid for AUTOG and RBLKnnnn requests. The named region must be defined in the REGIONS section of the EXTENT CONTROL file. The region defines a segment on a specific DASD volume for allocation.

DEVNO *raddr*

specifies a full volume minidisk. The use of this option is valid only in VM/ESA (ESA Feature) environment. *raddr* is the real device number of the full volume minidisk.

mode

modeV

specifies the access mode requested for the new minidisk, as seen by the owner. The valid access modes are listed as follows, and an optional V may be appended, signifying that virtual RESERVE/RELEASE processing is to be done.

Note: If mode is not specified, the resulting directory statement will not contain a mode entry and the CP default value, (W), will be in effect. There will be no access passwords. The owner can update these options directly, after the minidisk is created, using the MDISK operand.

Mode Definition

- R** Read-only (R/O) access is desired. The access will not be given if any other user has a link to the disk in write status.
- RR** Read-only (R/O) access is desired, even if another user has a link to the disk in write status.
- SR** Stable Read-only mode is requested.
- ER** Exclusive and Stable Read-only mode is requested.
- W** Write access is desired. The disk will not be accessible if any other user has a link to the disk in read or write status.
- WR** Write access is desired, but read-only access is acceptable if any other user has a link to the disk in read or write status.
- SW** Stable Write mode is requested.
- EW** Exclusive and Stable Write mode is requested.
- M** Multiple access is desired. A write link will be given to the disk unless another user already has write access to it, in which case the disk will not be accessible to you.
- MR** Write access is to be given to the disk unless another user already has write access to it, in which case read-only (R/O) access is given and message DEV xxxx FORCED R/O will be issued. This mode is the default.

MW specifies that write access is to be given to the disk unconditionally.

SM Stable multiple write access mode is requested.

BLKsize *nnnn*

specifies the block size to use when formatting the new extent. The valid values include:

512
1024
2048
4096
1K
2K
4K

If the BLKSIZE parameter is not specified, but the LABEL parameter is specified, the disk will be formatted with the default blocksize determined by the CMS FORMAT command. If neither BLKSIZE nor LABEL are specified, the disk will not be formatted.

LABEL *xxxxxx*

specifies the minidisk label to use when formatting the new extent. The labels are 1 to 6 alphanumeric characters.

If the LABEL parameter is not specified, but the BLKSSIZE parameter is specified, the disk will be formatted with a default label consisting of the minidisk address preceded by as much of the userid as will fit a 6 character label. If neither LABEL nor BLKSIZE are specified, the disk will not be formatted.

PWs

informs DirMaint that the following parameters are to be considered passwords.

readpass

is a password that allows a user other than the owner to get a read-only (R/O) link to the minidisk if the other user knows this 1 to 8 character password.

writepass

is a password that allows a user other than the owner to get a WRITE link to the minidisk if the other user knows this 1 to 8 character password.

multipass

is a password that allows a user other than the owner to get a MULTIPLE link to the minidisk if the other user knows this 1 to 8 character password.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Access mode and access passwords can be set and changed by the owner. If they are not entered with the AMDISK operand, the CP default will be access mode W, with no access passwords.

3. Due to the syntax of the CP MDISK statement, the use of access passwords requires that a mode be specified.
4. You may deliberately want to allocate a specific extent on a volume. If this extent would cross region boundaries or occupy a reserved area, you must have issued the EXTNCHK OFF operand before entering the AMDISK operand.
5. A PROFILE definition may not be the target of the AMDISK operand.
6. A specification of END for extent size is only valid for permanent (not T-DISK) fixed-allocation minidisks.
7. When the *valid* used on the AMDISK command does not exist in the EXTENT CONTROL file the volume will be allocated explicitly. The devtype value provided (3390, 3380) will be used as the default device type and DirMaint will use the default maximum size that is found in the DEFAULTS section of the EXTENT CONTROL file. For example, a 3390 device will default to a maximum allocatable cylinder value of 1113.

Examples

```
DIRM AMD 193 3390 15 4 VOL005
```

Adds a new 4-cylinder minidisk with a virtual device address of 193. It is to be allocated on a 3390 volume, *valid*= VOL005, at real cylinder number 15. The access mode will default to W (the CP default value), and no access passwords will be set.

Note: Extent validation may fail and cause the transaction to fail if the specified extent would overlap an existing minidisk.

```
DIRM AMD 192 X AUTOV 10 VOLXYZ
```

This requests automatic allocation for a 192 minidisk of 10 cylinders. The allocation is to be made on volume *valid*= VOLXYZ.

```
DIRM AMD 192 X VBLK4096 1800 VOLABC
```

This requests automatic allocation of a minimum of 1800 CMS 4K blocks on VOLABC as defined in the EXTENT CONTROL file. Examples: if VOLABC is a 3390, 10 cylinders will be allocated; if 3380, 12 cylinders; if 3375, 19 cylinders (1824 blocks total); or, 14400 FB-512 blocks for an FBA device.

```
DIRM AMD 23F X AUTOG 8 MSSVOLS
```

This is a request for a new 23F minidisk of 8 cylinders. This is quite like the AUTOV example, except that the allocation is to be made automatically on a region within group MSSVOLS. The group name, MSSVOLS, and all volumes within it, must be currently defined in the EXTENT CONTROL file.

```
DIRM AMD 199 FB-512 T-DISK 480
```

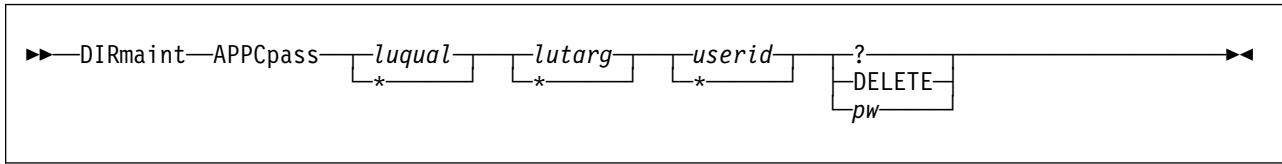
Requests a temporary minidisk of 480 blocks, on a 9336 or other FBA device be made available as virtual device address 199 for each terminal session.

```
DIRM AMD 123 9345 DEVNO 839 MR PWS RPASS WPASS
```

This assigns the real 9345 DASD volume at address 839, whatever the volume serial identification may be, starting at cylinder 0 for the entire volume as virtual address 123. This is similar to having a DEDICATE 839

123, except that other users may link to this DEVNO minidisk by virtue of the passwords.

APPCPASS



VM Systems: All

Purpose

The APPCPASS operand of the DIRMAINT command can be used to add a new APPCPASS statement to a general user's directory entry. This operand can also be used to delete, query, and change these statements in a general user's directory entry.

Operands

luqual

represents the 1 to 8 alphanumeric character string that identifies whether:

- the target resource is located in a TSAF collection (in which case the values are *IDENT or *USERID).
- the target resource resides in the SNA network (in which case you must specify the communication server name).

The asterisk (*) allows the APPCPASS operand to operate on all APPCPASS statements in the directory definition regardless of the LU qualifier.

lutarg

is the 1 to 8 byte character string that identifies the target LU name. If the LU qualifier is *IDENT, the target LU must be 0 (zero).

The asterisk (*) allows the APPCPASS operand to operate on all APPCPASS statements in the directory definition regardless of the LU target specified.

userid

is the 1 to 8 byte user ID of the target LU that authorizes the communication initiator to access resources located on the target LU.

The asterisk (*) allows the APPCPASS operand to operate on all APPCPASS statements in the directory definition regardless of the user ID.

? displays APPCPASS statements that meet the specified selection criteria.

DELETE

causes the APPCPASS statements that meet the specified selection criteria to be deleted from your directory definition.

pw is the 1 to 8 byte password that corresponds to the password of the *userid* at the target LU.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode, PROMPT) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For information regarding the use and specification of APPCPASS statements, see *z/VM: Planning and Administration*.
3. Because of the syntax of this operand, DELETE and ? cannot be used as passwords. If either of these is required as passwords, request that your administrator use DIRM GET and DIRM REPLACE.
4. The wild card character (*) may be used in one or more positions to qualify a set of APPCPASS statements to be queried, DELETED, or have their passwords changed. A specific APPCPASS statement is qualified using the keyword and the next three tokens.
5. It is better from a security standpoint not to enter access passwords on the command line. If the password is omitted, and the system prompts you for it, then it can be entered in the “display inhibit” mode. This may inhibit the use of complex passwords. If you enter the password directly on the command line, you should follow installation guidelines for security.

Directory Entry Samples

The following is a sample directory entry. Below the sample is a description of what would change in the entry due to several different invocations of the DIRM APPCPASS command operand.

The execution of each of these commands is contingent upon the users supplying the correct CP logon password for USERA when DirMaint prompts the user.

```

USER USERA LOGPASS 2M 2M G
ACCOUNT AAAAAAAA DIST
IPL CMS
APPCPASS LUQUAL1 LUTARG2 USERA PWA
APPCPASS LUQUAL3 LUTARG4 USERB PWB
APPCPASS LUQUAL3 LUTARG2 USERA PWA
APPCPASS LUQUAL3 LUTARG2 USERB PWB
CONSOLE 009 3215
SPOOL 00C 2540 R
SPOOL 00D 2540 P
SPOOL 00E 1403
LINK MAINT 190 190 RR
MDISK 191 3380 100 5 SYSRES M
    
```

Response messages containing the information on the last two APPCPASS statements in the sample directory entry will be sent to the USERA display if the following is issued by USERA:

```
DIRM APPC LUQUAL3 LUTARG2 * ?
```

In the sample directory entry, the password on all the APPCPASS statements will be changed to PWC if the following is issued by USERA:

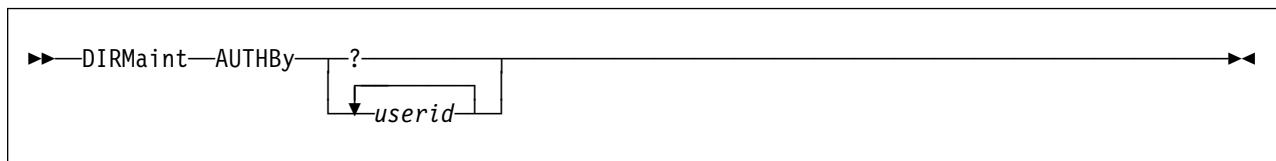
```
DIRM APPC * * * PWC
```

APPCPASS

In the above directory entry, the last three APPCPASS statements will be deleted if the following command is issued by USERA.

```
DIRM APPC LUQUAL3 * * DELETE
```

AUTHBY



VM Systems: All

Purpose

The AUTHBY operand of the DIRMAINT command is used to update the AUTHBY CONTROL file to allow the listed user ID to use the BYUSER prefix.

Operands

? requests that the users currently authorized to use the BYUSER prefix be shown.

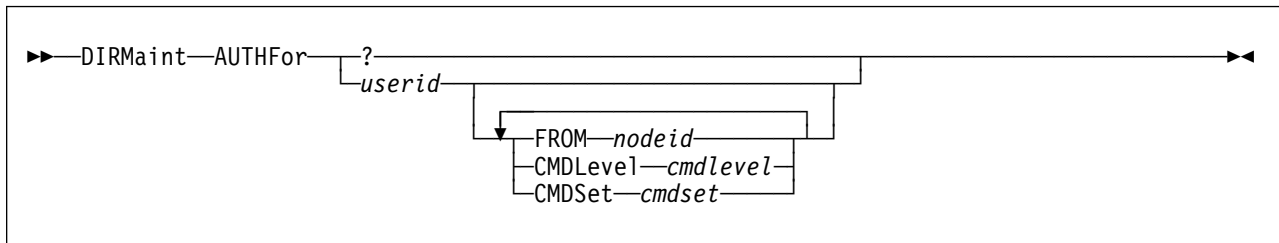
userid

is a list of one or more user IDs that should be added to the current AUTHBY CONTROL file.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

AUTHFOR



VM Systems: All

Purpose

The AUTHFOR operand of the DIRMAINT command is used to authorize one user to act on behalf of another user.

Operands

? requests that the current users authorized to act on behalf of the command issuer be reported.

userid

specifies the user ID that is being authorized to act on behalf of the command issuer.

FROM *nodeid*

specifies the node ID of the user that is being authorized to act on behalf of the command issuer.

CMDLevel *cmdlevel*

specifies the command level being authorized.

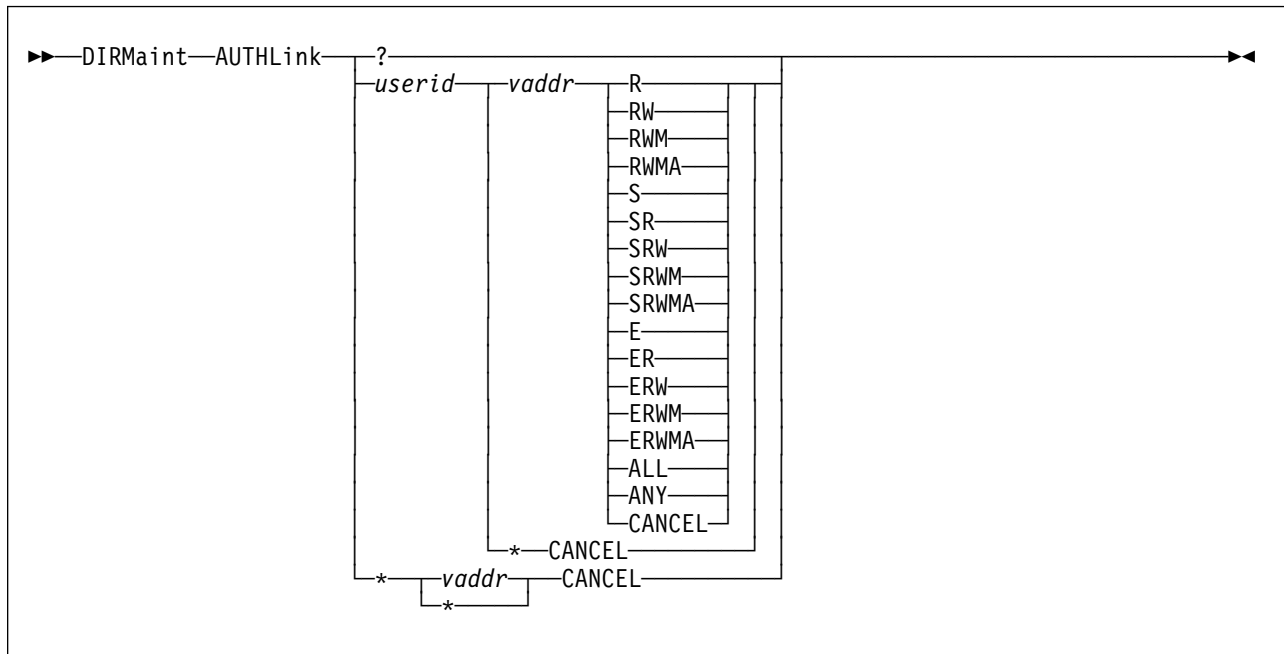
CMDSet *cmdset*

specifies the command set being authorized.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

AUTHLINK



VM Systems: All

Purpose

The AUTHLINK operand of the DIRMAINT command is used to allow a specified user or users to issue DIRM LINK commands without having to know the minidisk password for the disk being linked.

Operands

? requests that the current link authorizations be reported.

userid

***** specifies the user being granted the authorization. When used in place of the *userid* operand, the * value indicates that all previously granted authorities to link to the specified disk are to be cancelled. The * is used with the CANCEL keyword exclusively.

vaddr

***** specifies the address being authorized or cancelled. When used in place of the *vaddr* operand, the * value indicates that the specified user's authority to link to any of the owners disks is to be cancelled. The * is used with the CANCEL keyword exclusively.

R requests that Read links be authorized.

RW

requests that Read and Write links be authorized.

RWM

requests that Read, Write, and Multi links be authorized, excluding MW.

AUTHLINK

RWMA

requests that Read, Write, and Multi links be authorized, including MW.

S requests that Stable links be authorized.

SR

requests that Stable Read links be authorized.

SRW

requests that Stable Read/Write links be authorized.

SRWM

requests that Read, Write, and Multi links be authorized, excluding MW.

SRWMA

requests that Read, Write, and Multi links be authorized, including MW.

ERWM

requests that Read, Write, and Multi links be authorized, including MW.

E requests that Exclusive/Stable links be authorized.

ER

requests that Exclusive/Stable Read links be authorized.

ERW

requests that Exclusive/Stable Read/Multi links be authorized.

ERWM

requests that Read, Write, and Multi links be authorized, excluding MW.

ERWMA

requests that Read, Write, and Multi links be authorized, including MW.

ALL

allows any of the above modes to be specified.

ANY

allows any of the above modes to be specified.

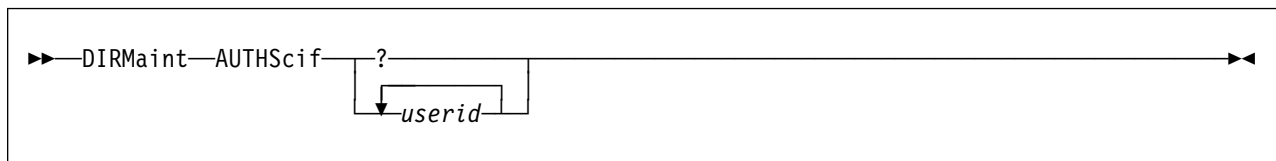
CANCEL

requests that the current link authorizations for the specified user/device combination be cancelled.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

AUTHSCIF



VM Systems: All

Purpose

The AUTHSCIF operand of the DIRMAINT command is used to allow a general user to authorize user IDs to act as secondary user IDs. The AUTHSCIF operand updates the AUTHSCIF CONTROL file, which controls users allowed to use the DIRMAINT SECUSER command.

Operands

? returns in a message a list of the user IDs that are authorized to use your user ID as a secondary user ID.

userid

is a list of one or more user IDs to be authorized to use your user ID as a secondary user ID.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The use of SECUSER, AUTHSCIF, DSECUSER and DROPSCIF commands are closely related. The *userid* must have granted the issuing user authority through the AUTHSCIF command to use them as a secondary console. An error message will result if this has not been done. For example, if the *userid* WATCHER chooses to become the secondary console of the disconnected user SERVER, this sequence of commands is needed:
 - a. WATCHER issues DIRM AUTHSCIF SERVER
 - b. SERVER issues DIRM SECUSER WATCHER

When the secondary user facility is no longer needed, one of two commands would be needed:

- a. WATCHER issues DIRM DSECUSER SERVER
- b. SERVER issues DIRM SECUSER OFF

If SERVER is not going to use WATCHER as a SCIF user again, the general authority can be cancelled if WATCHER issues:

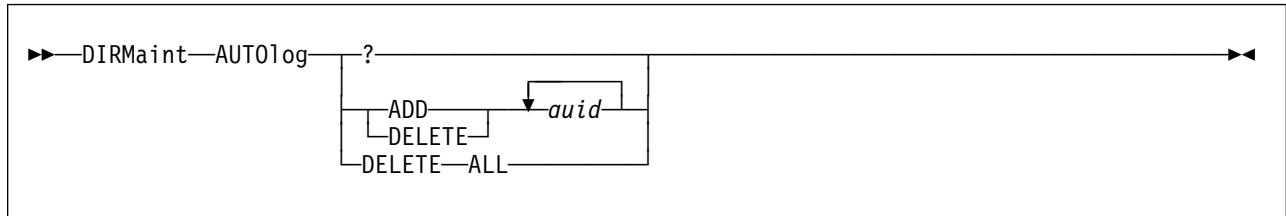
DIRM DROPSCIF SERVER

The design allows the SCIF target *userid* (WATCHER) to have overall control of the usage of SCIF, in case of SERVER produced SCIF output

AUTHSCIF

which the target was not able to handle. SERVER has sole control of the time it is turned on, but WATCHER must authorize it first. If the SECUSER command is issued before the nominated *userid* has granted authority with the AUTHSCIF command, an error message will result.

AUTOLOG



VM Systems: VM/ESA (ESA Feature)

Purpose

The AUTOLOG operand of the DIRMAINT command is used to add or delete a virtual machine name from an existing AUTOLOG or XAUTOLOG statement for a specified user's directory entry.

Operands

where:

? specifies that the existing AUTOLOG or XAUTOLOG statement is to be queried.

ADD

specifies that a user ID (*aid*) is to be added to an existing AUTOLOG or XAUTOLOG statement or a new statement is to be created.

DELETE

DELETE ALL

specifies that a user ID (*aid*) is to be deleted from the AUTOLOG or XAUTOLOG statement. If the ALL keyword is used the entire statement is deleted.

aid

specifies a 1 to 8 character name of the virtual machine authorized to AUTOLOG or XAUTOLOG the specified user ID.

Usage Notes

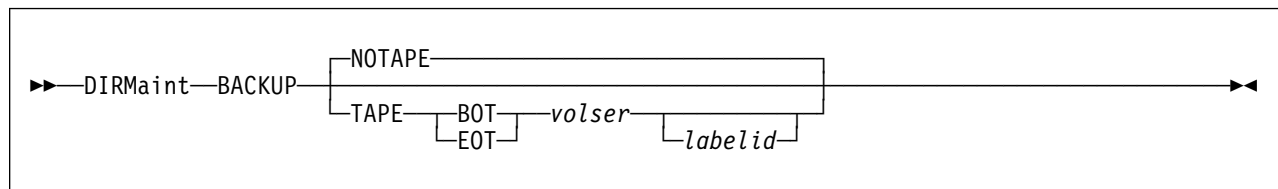
1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the AUTOLOG directory statement, see *z/VM: Planning and Administration*.
3. XAUTOLOG and AUTOLOG are synonymous but XAUTOLOG is the preferred directory keyword. Therefore, if the keyword AUTOLOG is encountered it is translated to XAUTOLOG.
4. A "DIRM XAUTOLOG ADD *aid*" may result in the creation of an XAUTOLOG statement, if there are no existing XAUTOLOG statements in the directory entry, or no more room on an existing XAUTOLOG statement. Similarly, a "DIRM XAUTOLOG DELETE *aid*" may result in the deletion of

AUTOLOG

an XAUTOLOG statement if there are no more *auids* on the existing XAUTOLOG statement.

5. Specifying DELETE for a user ID not on an AUTOLOG or XAUTOLOG statement will generate an error message. Specifying ADD for a user ID already on an AUTOLOG or XAUTOLOG statement will also generate an error message.
6. A maximum of eight user IDs may be specified via AUTOLOG or XAUTOLOG.
7. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

BACKUP



VM Systems: All

Purpose

The BACKUP operand of the DIRMAINT command is used by staff members to manually trigger the normally automatic backup procedure due to an unusual or emergency condition.

Operands

NOTAPE

requests that a backup be done to the DirMaint primary backup disk. A monolithic copy of the current directory and all control files are placed on to this disk.

TAPE

requests that a backup be done to disk and then to a tape device.

BOT

EOT

controls the positioning of the tape device when tape backups are being made. BOT specifies that the backup will be made at the Beginning Of the Tape (if *labelid* is also specified, the backup will be made after the tape label). EOT specifies that the backup will be made at the End Of the Tape.

volser

specifies the volume serial of the tape, to be displayed in the message to the operator.

labelid

specifies the tape label, for verification that the correct tape has been mounted.

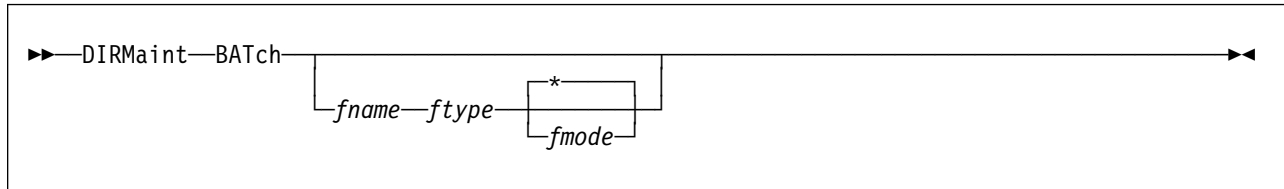
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The backup procedure is executed automatically each day at the time specified in the DIRMAINT DATADVH file. If the system is down at the time specified in the DIRMAINT DATADVH file, backup will be automatically executed during DirMaint IPL.
3. The difference between the BACKUP operand and the USER BACKUP operand is that the USER BACKUP operand causes regeneration of the USER BACKUP file only. The BACKUP operand invokes the full backup procedure that backs up the auxiliary control files as well.

BACKUP

4. Use of BACKUP TAPE requires the BACKUP_TAPE_MOUNT_EXIT be defined in the CONFIG* DATADVH file(s). The IBM supplied sample (DVHXTAPE EXEC) uses simple messages to the operator. The exit may be customized to interface with a variety of tape library management programs.

BATCH



VM Systems: All

Purpose

The BATCH operand of the DIRMAINT command allows users to submit any number of commands to the DIRMAINT service machine in one file. The commands will execute in a background work queue, as opposed to the interactive work queue, from which individual user commands are executed. DIRMAINT queries the background queue for work only when the interactive work queue is empty.

Operands

fname

specifies the file name of the batch file.

ftype

specifies the file type of the batch file.

fmode

* specifies the file mode of the batch file.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. After entering DIRM BATCH (without operands), the user is placed in an XEDIT session, from which the user can enter any number of DIRMAINT commands, one per line. When command input is complete, the file is sent to DIRMAINT for processing. DIRMAINT sends messages when it opens the batch to begin work and when work on each command is begun and finished. If the submitting user ID is not logged on at that time, DIRMAINT logs these messages in the usual LOG file for that user.
3. The commands entered into the batch file while in the XEDIT session should not begin with the normal DIRM or DIRMAINT keyword as it does when a non-batch command is being issued. For example, if the user wishes to issue the command:

```
DIRM LINK USERA 111 222 DELETE
```

as one of the commands in a batch, the record in the batch file should be:

```
LINK USERA 111 222 DELETE
```

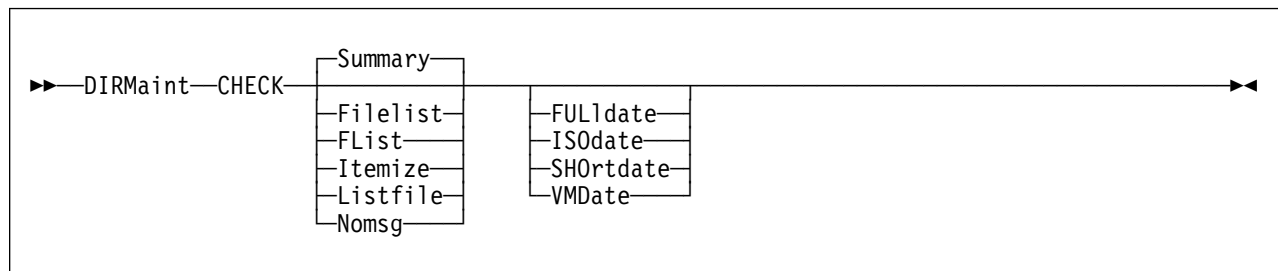
not

```
DIRM LINK USERA 111 222 DELETE
```

BATCH

4. When the resulting file from a BATCH command is submitted for processing, it is also saved on the invoker's A-minidisk. The file is retrieved the next time the DIRM BATCH command is entered.
5. Transactions such as commands and operands entered into the batch file can not involve the transfer of any bulk data files. Operands such as ADD (unless the LIKE prototype facility is being used), MAIL, BATCH, FILE, PWMON LOCKOUT, PWMON REPLACE CONTROL, and REPLACE are examples of such transactions.

CHECK



VM Systems: All

Purpose

The CHECK operand of the DIRMAINT command is used to determine if there are extraneous DirMaint product files on any accessed disk or SFS directory.

Operands

Filelist

Replaces the ITEMIZE diagnostic messages (DVH1371) with an actual EXEC FILELIST command to enable further manual checking for potential problems.

FList

Replaces the ITEMIZE diagnostic messages (DVH1371) with an FLIST command to enable further manual checking for potential problems.

If the FLIST MODULE is not installed (from either the VM/Interactive Productivity Facility (VM/IPF), or the VM/ESA CMS Utilities Feature, or equivalent) then an EXEC FILELIST command is used instead.

Itemize

Produces a detailed diagnostic message (DVH1371) for each extraneous file found in your search order.

Listfile

Replaces the ITEMIZE diagnostic messages (DVH1371) with a LISTFILE command to provide further diagnostic information on potential problems. This information includes the date and time stamps and the disk label for each extraneous file found in your search order.

Note: The output will reside in DVHUCHK OUTPUT A.

Nomsg

Suppresses all diagnostic messages. The only remaining indication of any potential problems is a non-zero return code.

Summary

Suppresses the itemized diagnostic messages (DVH1371) but produces a summary diagnostic message (DVH1372) showing the total number of extraneous files found.

FULLdate

Uses the FULLDATE option on the FILELIST or LISTFILE commands. This is valid for VM/ESA 2.2.0 and follow-on releases.

CHECK

ISOdate

Uses the ISODATE option on the FILELIST or LISTFILE commands. This is valid for VM/ESA 2.2.0 and follow-on releases.

SHORtdate

Uses the SHORTDATE option on the FILELIST or LISTFILE commands. This is valid for VM/ESA 2.2.0 and follow-on releases.

VMDate

Uses the VMDATE option on the FILELIST commands. This is valid for VM/ESA 2.2.0 and follow-on releases.

Usage Notes

1. DirMaint prefix operands do apply to this command. They may be accepted but will be ignored.
2. The CHECK command is completely handled within the virtual machine of the command issuer. No transaction is sent to the DIRMAINT service machine, therefore no password is required or requested for authentication, even if you have NEEDPASS set to YES.
3. The CHECK command is not valid within a BATCH command file.
4. The return code from the CHECK command is the number of extraneous files found.
5. This command will create a scratch file called DVHUCHK OUTPUT A. Except for the LISTFILE option, this scratch file will be erased when the command is complete. The FLIST option will also create and erase the CMS EXEC A file.
6. This command is used to investigate the cause of problems. Since duplicate files don't cause problems to the DIRMAINT machine, extraneous or duplicate files are not reported.
7. For more information about the FULLDATE, ISODATE, SHORTDATE, or VMDATE operands, use a HELP CMS FILELIST or HELP CMS LISTFILE command, see *z/VM: CMS Command Reference*.

CHKSUM

▶▶—DIRMaint—CHKsum————▶▶

VM Systems: All

Purpose

The CHKSUM operand of the DIRMAINT command is used to verify the integrity of the source directory.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The act of generating or regenerating the directory initializes the checksum values. Once the directory source is initialized, it contains individual checksums for each profile and user ID. A checksum is a logical arithmetic sum of the contents of a directory entry. They are used to guard against partial loss of data due to DASD I/O errors. Checksums are recomputed each time a directory entry is altered, and compared against the stored value. If a mismatch is detected, your system administrator is informed and should take corrective action. These checksums are stored in a special record that follows each profile and user definition. These checksums are maintained by DirMaint as each transaction modifies the source directory.
3. Once the directory has been initialized, it should not be edited directly. Direct editing invariably introduces checksum errors, possibly for every entry if default serialization is allowed to take place. An editor may only be used as part of the cycle of transactions (GET operand, edit, REPLACE operand), when applied to a single directory entry at a time.

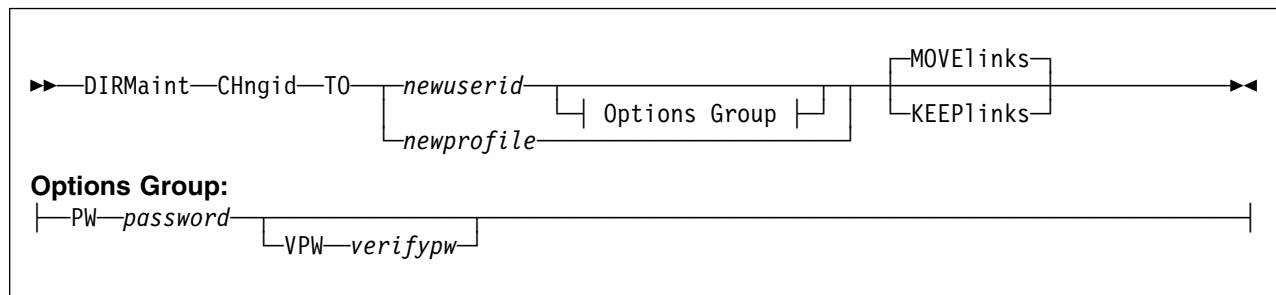
To make multiple updates to the entire directory (updates that may affect multiple users) the following procedure is recommended and should be done as a BATCH file, issue DIRM BATCH (the DIRM prefix must be removed from each command) then:

- a. Disable updates to the source directory (DIRM DISABLE)
- b. Ensure you have the latest copy of the directory (DIRM USER BACKUP)
- c. Receive the monolithic backup (DIRM SEND USER BACKUP G)
- d. Receive the file to your A-minidisk and use your editor to make the changes.
- e. Replace the user input file (DIRM FILE USER BACKUP A USER INPUT E) on the DIRMAINT server.
- f. Erase the USER DIRECT file (DIRM CMS ERASE USER DIRECT E)
- g. Initialize the directory structure (DIRM CP IPL CMS PARM AUTOOCR)
- h. Enable updates to the source directory (DIRM ENABLE)

CHKSUM

4. The CHKSUM operand reads through the source directory, generating a new checksum for each profile and user definition and comparing the newly computed value with the stored value. Checksum errors will be reported but not corrected.
5. If the number of definitions with faulty checksums is small, they can be corrected by performing a GET/Edit/REPLACE cycle on each definition in error. REPLACE processing will automatically regenerate the checksum value for each definition being replaced in the source directory. If there are a large number of definitions in error, consider initializing the directory from a monolithic copy (USER INPUT). New checksum values will be automatically recalculated for all definitions.

CHNGID



VM Systems: All

Purpose

The CHNGID operand of the DIRMAINT command is used to change the user ID of an existing user or the name of an existing profile.

Operands

TO

is a required keyword.

newuserid

specifies the new user ID name.

newprofile

is the new profile entry name.

PW *password*

specifies a new CP logon password that will be assigned to *newuserid*.

VPW *verifypw*

specifies the same password again for typographical verification

KEEPlinks

requests that the links the devices owned by this user ID not be changed to reflect the new name.

MOVElinks

requests that the links the devices owned by this user ID be changed to reflect the new name.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. This operand is equivalent to the following series of operations:
 - a. GET (with the NOLOCK option) the existing user ID or profile. The existing user ID or profile DIRECT file will be sent to your reader.

Note: You must use the NOLOCK option when performing this step to avoid a PURGE failure problem when performing step 2c on page 54.

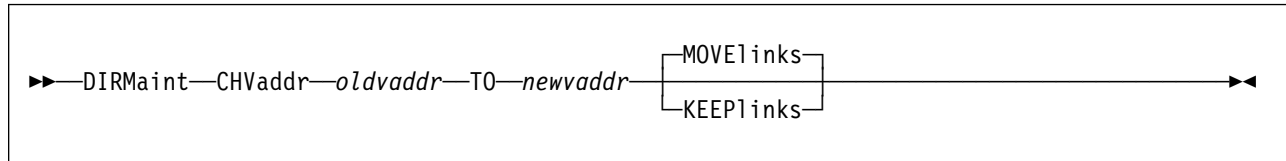
- b. Read in the existing user ID or profile DIRECT file. Use the editor to change the user ID in the USER control statement to *newuserid* or the profile name in the PROFILE control statement to *newprofile*. Rename the file to *newuserid* DIRECT or *newprofile* DIRECT.
 - c. PURGE the old user ID or profile with the NOCLEAN option.
 - d. ADD *newuserid* or *newprofile*.
3. You cannot change the name of a profile if any INCLUDE statements invoke it. If you try, an error message will be issued and the change will be rejected. User entries using the subject profile will be indicated in the returned messages. If you want to use CHNGID to change a profile name that is the target of INCLUDE statements, you must perform the following steps in order:
- a. Use GET, edit, and REPLACE to delete all the INCLUDE statements that invoke the profile in question, or to change them so that they invoke another profile. An alternative would be to use the DIRM INCLUDE command. This command allows the deletion of INCLUDE statements.
 - b. Use CHNGID to change the profile name.
 - c. Use GET, edit, and REPLACE to replace the INCLUDE statements or to change them so that they invoke the profile by its new name. An alternative would be to use the DIRM INCLUDE command. This command allows the creation and alteration of INCLUDE statements.

It is easier to change a profile name this way:

- a. Use ADD to add a copy of the profile with the new name.
 - b. Use GET, edit, and REPLACE to change all the INCLUDE statements that invoke the profile in question so that they invoke the profile by its new name. An alternative would be to use the DIRM INCLUDE command. This command allows the creation and alteration of INCLUDE statements.
 - c. Use PURGE to delete the old copy of the profile.
4. The CHNGID operand can also change the user's logon password.
5. Attempts to specify a password when changing a PROFILE result in the transaction being rejected.
6. If an account record exists in the entry to be changed, the account number verification exit (ACCOUNT_NUMBER_VERIFICATION_EXIT= in the configuration file) is called. The exit may reject the account numbers. If this is done the transaction is rejected.
7. CHNGID processing will invoke the logon password user exit facility (if one is in use at the installation), even if there is no change in the logon password. This is done to advise the exit of the change in the user ID name. The rationale for this action is that an installation may be using the exit to maintain installation-defined files. If a password change is involved, the exit may subject the proposed password to the normal installation-defined checks.
8. Any links to this device are changed to match the *newuserid* and *newprofile*. This is done as a batch request. Because the execution of such work may be deferred, it is possible that one or more of the linking

user IDs might manually change their own LINK statements (using the LINK operand) before the low priority work is executed. If this occurs, the low priority command to change the link will fail when selected for execution. This error condition can be safely ignored.

CHVADDR



VM Systems: All

Purpose

The CHVADDR operand of the DIRMAINT command is used to change the virtual device address of an existing minidisk.

Operands

oldvaddr

is the current virtual device address.

TO

is a required keyword.

newvaddr

is the virtual device address that the minidisk address is to be changed to.

KEEPlinks

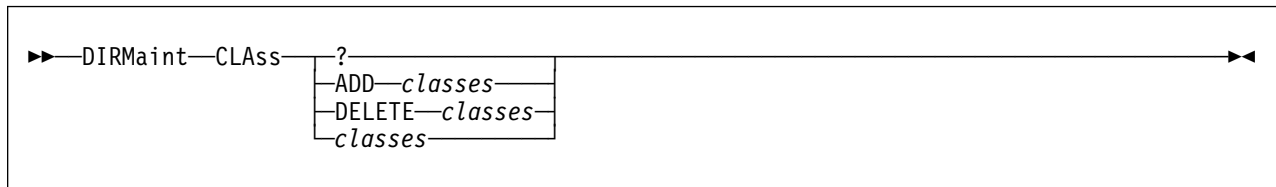
requests that links to this device not be changed to reflect the new virtual address.

MOVElinks

requests that links to this device be changed to reflect the new virtual address.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The control statement in the directory entry for user ID is updated to reflect the new *vaddr*.
3. Any links to this device are changed to match the new *vaddr*. This is done as a batch request. Because the execution of such work may be deferred, it is possible that one or more of the linking user IDs might manually change their own LINK statements (using the LINK operand) before the low priority work is executed. If this occurs, the low priority command to change the link will fail when selected for execution. This error condition can be safely ignored.

CLASS


VM Systems: All

Purpose

The CLASS operand of the DIRMAINT command enables you add, delete, replace, or query the current class value for a user directory entry.

Operands

? allows you to query your current class values.

ADD *classes*

DELETE *classes*

specifies the new class values to be added to the existing class values, or the class values that are to be deleted from the existing class values.

classes

when specified without the ADD or DELETE keyword, the new class values replace the current class information in the user directory entry.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For a complete description of the CLASS directory statement and a discussion on how it relates to the USER statement, see *z/VM: Planning and Administration*.
3. An attempt to add a class value when it already exists on the user statement will be rejected.
4. An attempt to delete a class value from the user statement when that value does not exist will be rejected.

CLEANED

CLEANED

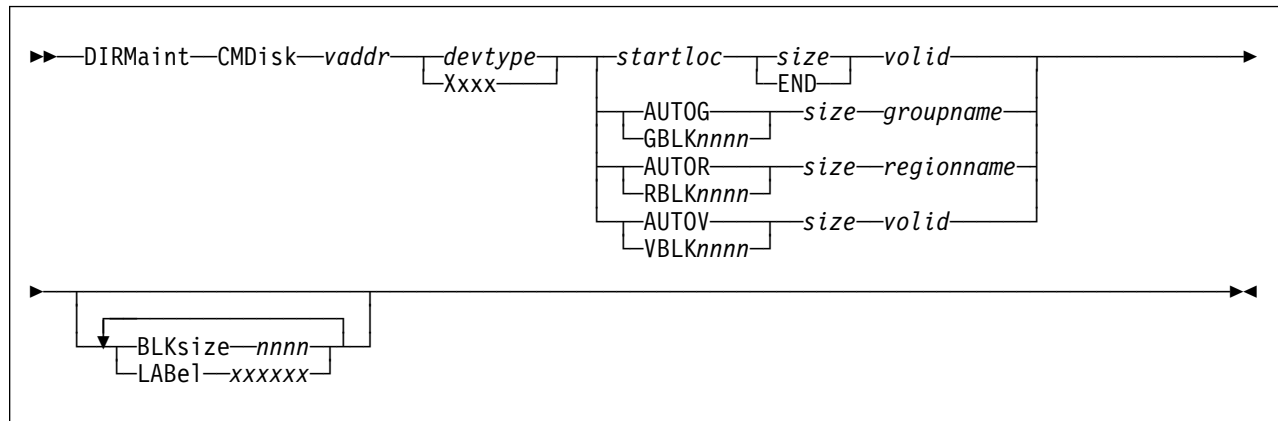
▶▶—DIRMaint—CLEaned—*vaddr*—————▶◀

VM Systems: All

Purpose

The CLEANED operand of the DIRMAINT command is obsolete. This operand is tolerated to remain compatible with earlier releases of DirMaint. Its usage will not cause an error, but its function is obsolete. If issued, a warning message is returned indicating that the function of the operand is no longer applicable.

CMDISK



VM Systems: All

The CMDISK operand of the DIRMAINT command is used to change an existing minidisk in terms of residency, size, and extent occupied.

The disk must be formatted as a CMS disk and must not be RECOMPed or RESERVED. DirMaint will format the new extent and move existing files to the new extent. If no files are involved, refer to DIRM RMDISK to alter an existing extent.

See Usage Note 9 on page 62 for DirMaint-owned minidisks.

Operands

vaddr

is the virtual device address of the minidisk.

devtype

is the device type of the volume to which the minidisk is assigned. Otherwise, if automatic allocation is used, specify the device type as Xxxx. Automatic allocation options include:

- AUTOG and GBLK
- AUTOR and RBLK
- AUTOV and VBLK

operations. The real device type will be taken from the REGIONS section in the EXTENT CONTROL file for that volser.

startloc

is a decimal cylinder relocation factor or a decimal FB-512 device block number that specifies the beginning of the minidisk extent corresponding to cylinder 0 or block 0 of the user's minidisk.

AUTOG

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a volume within a specified group. The group must have been defined in the EXTENT CONTROL file, and the regions within the group must also be defined.

GBLKnnnn

is functionally equivalent to AUTOG except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: GBLK512, GBLK0512, GBLK1024, GBLK2048, GBLK4096, GBLK1K, GBLK2K, and GBLK4K. In addition, GBLK800 or GBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

If the existing disk is permanently allocated and has already been formatted for CMS use, the new extent will actually be formatted with the same block size as the existing disk. However, DirMaint will use the block size specified to compute the number of CKD cylinders necessary to satisfy the request.

AUTOR

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a specified region or volume. The region must have been defined in the EXTENT CONTROL file.

RBLKnnnn

is functionally equivalent to AUTOR except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: RBLK512, RBLK0512, RBLK1024, RBLK2048, RBLK4096, RBLK1K, RBLK2K, and RBLK4K. In addition, RBLK800 or RBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

AUTOV

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a specified volume. The volume must have been defined in the EXTENT CONTROL file.

VBLKnnnn

is functionally equivalent to AUTOV except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: VBLK512, VBLK0512, VBLK1024, VBLK2048, VBLK4096, VBLK1K, VBLK2K, and VBLK4K. In addition, VBLK800 or VBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

If the existing disk is permanently allocated and has already been formatted for CMS use, the new extent will actually be formatted with the same block size as the existing disk. However, DirMaint will use the block size specified to compute the number of CKD cylinders necessary to satisfy the request.

size

is a 1 to 8 digit decimal number specifying the size of the new minidisk in CKD cylinders or FBA blocks. For allocation requests involving block sizes, allocation will be converted to the exact number of FB-512 blocks for FBA devices or to the minimum number of cylinders needed to contain the specified number of CMS blocks for CKD devices.

END

specifies that the minidisk will extend from the assigned starting cylinder to the end (last cylinder) of the volume in which it is defined. This parameter is valid for fixed allocation only.

valid

is the volume label of the DASD volume, 1 to 6 alphanumeric characters, on which the minidisk is to be allocated, or the reserved identifier of &SYSRES. This applies to fixed allocation (where a specific starting cylinder or block number is specified) and to automatic allocation on a volume (AUTOV, VBLKnnnn).

groupname

specifies the name of the DASD allocation group from which DirMaint will pick the *volser* and starting CKD cylinder or FBA block. Groups consist of one or more specific *regions*. This is valid for AUTOG and GBLKnnnn requests only. Groups must be defined in the EXTENT CONTROL file, and can be 1 to 8 alphanumeric characters.

regionname

is the name associated with the *region*, 1 to 8 alphanumeric characters, on which the minidisk is to be allocated. This is valid for AUTOR and RBLKnnnn requests. The named region must be defined in the REGIONS section of the EXTENT CONTROL file. The region defines a segment on a specific DASD volume for allocation.

BLKsize nnnn

specifies the block size to use when formatting the new extent. The valid values include:

512
1024
2048
4096
1K
2K
4K

If the BLKSIZE parameter is not specified, but the LABEL parameter is specified, the disk will be formatted with the default blocksize determined by the CMS FORMAT command. If neither BLKSIZE nor LABEL are specified, the disk will not be formatted.

BLKsize nnnn

specifies the block size to use when formatting the new extent. The valid values include:

512
1024
2048
4096
1K
2K
4K

If the BLKSIZE parameter is not specified, the new extent will be formatted with the same blocksize as the old extent.

LABel xxxxxx

specifies the minidisk label to use when formatting the new extent. The labels are 1 to 6 alphanumeric characters.

If the LABEL parameter is not specified, the new extent will be given the same label as the old extent.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. It is not advisable to define a minidisk starting at real cylinder 0 unless it is to be used by OS ISAM. If it is to be used by OS ISAM, it must begin at real cylinder 0. If you do assign a minidisk beginning at real cylinder 0, the user who owns it must be aware that the minidisk label is the real label that the user and the system use to identify the disk. CP-owned volumes must not have minidisks beginning at real cylinder 0. Generally, CKD DASD extents should begin with cylinder 1 or higher, and FBA DASD extents should begin with block 32 or higher.
3. In using this operand be aware that another user may be linked to the minidisk being changed. It is your responsibility to have that link removed and re-established after the change if necessary.
4. This operand is not used to change the access mode or access passwords. The owner of the minidisk can use the MDISK operand to do that directly.
5. The CMDISK operand can be used to change the size (number of cylinders or blocks) of a minidisk. This requires that a new extent be allocated, the files on the old extent copied to the new, and finally that the old extent be released. Data movement is not always automatic. See Usage Note 8.
6. The CMDISK operand can also be used to move the extent to a new residence without changing the size.
7. Converting a T-DISK minidisk to a permanent allocation requires the use of DIRM RMDISK.
8. When data must be moved from the old extent to the new extent, the move will be coordinated by a DIRMAINT virtual machine. The actual movement of data is done automatically by a DATAMOVE virtual machine.
 - a. The new extent is allocated, but attached to a user defined as
 DATAMOVE_MACHINE= *userid node sysaffin*
 in the CONFIG* DATADVH file. The role of this user is to do the copying and reformatting operations. Note that it is possible to define several DATAMOVE machines.
 - b. The user's original minidisk is also transferred to a user defined as
 DATAMOVE_MACHINE= *userid node sysaffin*
 in the CONFIG* DATADVH file. The role of this user is to do the copying and reformatting operations.
9. Do not attempt to move DirMaint minidisks or RACF® owned minidisks with the CMDISK operand. This must be done manually using this procedure:

 For each minidisk to be moved, use the AMDISK function to allocate a new extent as desired for the minidisk to be moved. Use a virtual address that you can relate to the basic MDISK. For example, if you are moving the 1DB, allocate a new 5DB.

After the newly-allocated extents are really available:

- a. Use the OFFLINE operand. This prevents updating the online directory during the next few steps.
- b. Use the GET operand. Edit the returned file (DIRMAINT DIRECT) and swap the virtual addresses of the minidisks.

Original Extent:

```
MDISK 1DB 3380 123 005 vo1001 MR
```

New Extent:

```
MDISK 5DB 9345 487 006 vo1016 MR
```

Change as follows:

Original Extent:

```
MDISK 5DB 3380 123 005 vo1001 MR
```

New Extent:

```
MDISK 1DB 9345 487 006 vo1016 MR
```

- c. Use the REPLACE operand. This updates the USER DIRECT file to map DirMaint's minidisks to the new extents. The OFFLINE operand prevents it from being placed online.
- d. Use the SHUTDOWN operand. DirMaint logs off, but the online directory still maps DirMaint's minidisks to the original extent(s). The source directory, however, reflects the newly-allocated extents as current.
- e. Log on to the DIRMAINT service machine.
- f. Format the newly-allocated minidisk(s), for example, 5DB.
- g. Access the original 1DB as A, and copy all files from the original 1DB (A) to the newly-allocated and formatted 5DB. Do this for all other minidisks being moved at this time. When all minidisks have been copied to the newly-allocated extents, continue with the next step.
- h. Manually invoke the system command to update the object directory (using DIRECT for VM/ESA (370 Feature), DIRECTXA otherwise). The source directory already contains the new minidisks. The new online directory contains mapping to the newly-allocated minidisks. These new extents are mirror images of the original extents, as shown in the example above.
- i. Log off, then as an authorized user, AUTOLOG the DIRMAINT service machine. The DIRMAINT machine will be operational using the new extents. When you are sure the DIRMAINT machine is operating correctly, use the ONLINE operand to re-enable execution of DIRECT. Then use the DMDISK operand to delete the original extents (now 5DB, and so on).

The purpose of this procedure is to preserve the integrity of DirMaint as an operational entity, and to ensure that any unrelated directory changes made during the time span required to perform the first few steps are preserved.

10. The CMDISK operand cannot be used to change an existing DEVNO, T-DISK, or V-DISK minidisk. The CMDISK operand will be rejected if the existing minidisk being changed is a DEVNO, T-DISK, or V-DISK minidisk.

The RMDISK operand may be used to change the minidisk definition, but does not cause any data movement to take place. Alternatively, DMDISK, followed by AMDISK, or GET and REPLACE, may be used to modify the definition of a DEVNO, T-DISK, or V-DISK minidisk.

11. If the minidisk is not a CMS formatted disk or if it is either RESERVED or RECOMPed, use AMDISK to add a new disk of the required type and size, have the owning user copy the existing data to the new disk then use DMDISK with the KEEPLINKS option to destroy the old extent. Follow this by a CHVADDR to reassign the new disk to the old disks address.

Canceling a DIRMAINT CMDISK Request

Many of the situations involving manual intervention in previous releases of DirMaint have been handled automatically in this release. Situations that result in DATAMOVE's inability to copy data (Non-CMS formatted minidisk or nonformatted minidisk for example) cause DIRMAINT to automatically rollback the CMDISK request and return the extent to the original owner.

There are situations where a user-initiated cancel of a CMDISK request may be required. If active links exist to the source of a CMDISK, DATAMOVE is forced to wait until these links are detached. When the links are detached the DATAMOVE machine will continue with its operation. Should the user wish to cancel the transaction while this wait is in progress the following steps should be used.

1. Find the work unit number associated with your transaction.

This can be done using the DIRM STATUS and DIRM QUERY commands. Each workunit contains information on who issued the command and what the command was.

2. Use the DIRM WORKUNIT command to CANCEL the appropriate workunit.

This will cause DATAMOVE to give up waiting on the active links and return control of the work unit to DIRMAINT. DIRMAINT will rollback (return the extent to the user and release all locks) and deallocate the workunit.

Should a work unit fail for some reason, DirMaint is designed to automatically rollback the transaction for the most common failures. Failures considered too dangerous for automatic rollback are left for administrative review and action. For more information on automatic and manual rollback of workunits, see *Directory Maintenance VM/ESA: Tailoring and Administration Guide*.

Examples

```
DIRM FOR JONES CMD 193 FB-512 1672000 880 VOL005
```

Changes the 193 minidisk belonging to JONES, from whatever and wherever it is, to 880 blocks. It is to be allocated on a 9336 or other FBA volume, valid=VOL005, at real block 1672000. The current access mode and access passwords, if any, remain unchanged.

```
DIRM FOR BILL CMD 192 X AUTOV 10 VOLXYZ
```

Specifies that the 192 minidisk belonging to user ID BILL be changed to 10 cylinders. The allocation is to be made on volume valid=VOLXYZ.

```
DIRM FOR STEVE CMD 186 3390 VBLK512 1000 V3390A
```

This command tells DirMaint to change STEVE's existing 186 minidisk, whatever its existing device type and location to 2 cylinders (1470 blocks total) on 3390 volume V3390A. The new disk will have the same CMS block format as the old disk: 990 blocks if formatted with BLKSIZE 1024, 630 blocks if formatted with BLKSIZE 2048, or 360 blocks for BLKSIZE 4096. For a non-CMS disk, the administrator must copy the data and use the appropriate DIRMAINT commands to let DirMaint know when to make the new extent available to the user. But the size is still computed the same: 1000 blocks requested, divided by 735 512-byte blocks per 3390 cylinder, equals 1.36 cylinders, which is increased to 2 cylinders to satisfy the minimum requirement.

```
DIRM FOR HARRY CMD 23F X AUTOG 8 MSSVOLS
```

Requests that HARRY's 23F minidisk be changed to 8 cylinders. This is like the AUTOV example, except that the allocation is to be made automatically on a region within group MSSVOLS. The group name, MSSVOLS, and all regions within it, must be currently defined in the EXTENT CONTROL file.

CMS

►►—DIRMaint—CMS—*command*—————◄◄

VM Systems: All

Purpose

The CMS operand of the DIRMAINT command is used to execute a CMS command string on the DIRMAINT server.

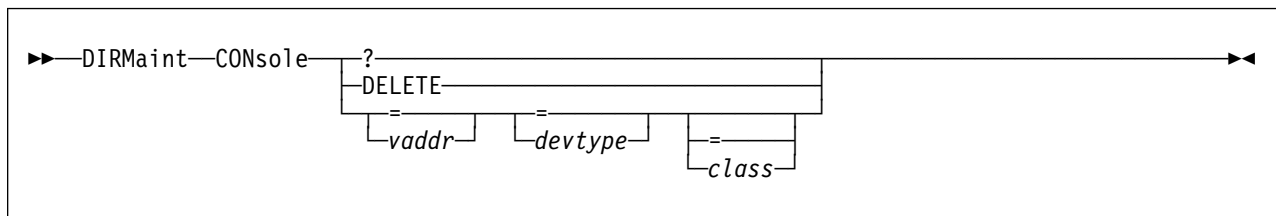
Operands

command
specifies the command to be executed on the DIRMAINT server.

Usage Note

You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

CONSOLE



VM Systems: All

Purpose

The CONSOLE operand of the DIRMAINT command is used to create, delete, or query the directory CONSOLE statement.

Operands

? specifies that the directory CONSOLE statement is to be queried.

DELETE

specifies that the directory CONSOLE statement is to be deleted.

= specifies the value currently contained in the position is to remain.

vaddr

specifies the virtual address of the directory console.

devtype

specifies the device type. Valid device types are 1052, 3210, 3215, and 3270.

class

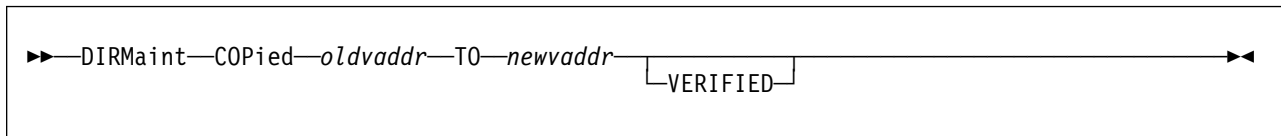
specifies the 1 character spooling class.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For a complete description of the CONSOLE directory statement, see *z/VM: Planning and Administration*.
3. To alter the ID of the secondary console, see "SECUSER" on page 214 .
4. The equal (=) parameter implies that the token in that position already exists. If the effective CONSOLE statement does not have a parameter in the position denoted by the equal sign, the request will be rejected.
5. The function performed by Release 4 has been renamed to GETCONSOLE, for Release 5 command level 150A. For more information, see "GETCONSOLE" on page 115 . In level 140A the CONSOLE command performs the same function as in Release 4. In level 150A the CONSOLE command changes the device characteristics of the virtual machine's CONSOLE directory statement.

COPIED

COPIED




VM Systems: All

Purpose

The COPIED operand of the DIRMAINT command is obsolete. This operand is tolerated to remain compatible with earlier releases of DirMaint. Its usage will not cause an error, but its function is obsolete. If issued, a warning message is returned indicating that the function of the operand is no longer applicable.

CP



▶▶—DIRMaint—CP—*command*—◀◀

VM Systems: All

Purpose

The CP operand of the DIRMAINT command is used to execute a CP command string on the DIRMAINT server.

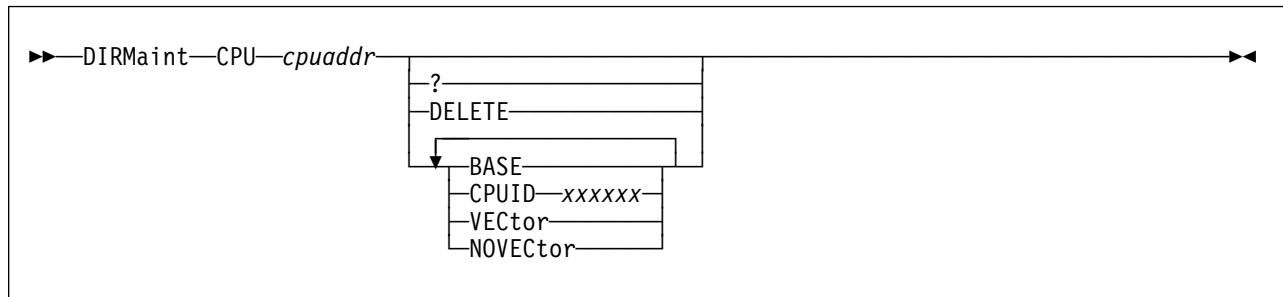
Operands

command

specifies the command to be executed on the DIRMAINT server.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

CPU


VM Systems: VM/ESA (ESA Feature)

Purpose

The CPU operand of the DIRMAINT command allows the general user to add, change, delete, or query the central processing unit (CPU) statement in a directory entry.

Operands

cpuaddr

specifies the address of the virtual processor being defined, changed, or deleted. The processor address can be any hexadecimal value from 0 to 3F.

? issues a query against the specified CPU statement.

DELETE

deletes the specified CPU statement from the directory entry.

BASE

specifies that the CPU identified on this statement is defined as the base CPU.

CPUID *xxxxxx*

specifies the processor ID number that is stored in bits 8 to 31 of the CPU ID and is returned in response to the STIDP instruction.

VECTor

NOVECTor

specifies whether a Vector Facility is assigned for the virtual CPU at logon.

Usage Notes

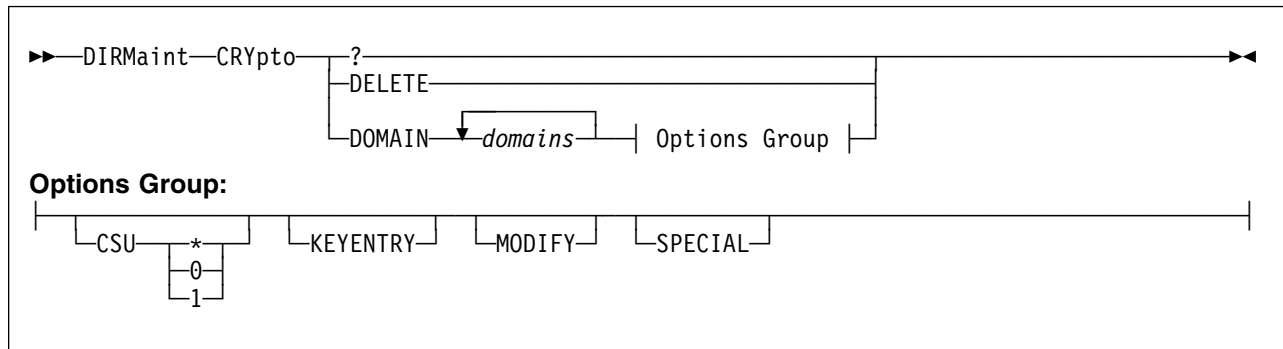
1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the CPU directory statement, see *z/VM: Planning and Administration*.

3. The CPU operand is equivalent to the SETCPU operand except that CRYPTO, DEDICATE and NODEDICATE are not valid on the CPU operand.
4. Successful execution of the DIRMAINT CPU operand depends on the MACHINE directory statement setting defining the maximum number of virtual processors allowed to be defined (indicated by the *mcpu* value). The following rules apply:
 - Modification of an existing CPU statement is always valid. However, only one CPU statement can be given the BASE attribute.
 - Deletion of an existing CPU statement is always valid.
 - If a MACHINE statement does exist and the *mcpu* field contains a value, this value is the maximum number of allowable CPU statements.
 - If a MACHINE statement does not exist, or does not specify an MCPU value, only one CPU may be defined.

If any of these rules are violated, CPU operand processing will terminate and an appropriate error message is issued.

5. A new CPU statement is always added after all existing CPU statements in your directory entry unless your installation has chosen to sort by device addresses. The order of CPU statements determines the assignment of real processors to virtual CPUs.

CRYPTO



VM Systems: VM/ESA (ESA Feature)

Purpose

The CRYPTO operand of the DIRMAINT command is used to add, delete, or query the current CRYPTO statement specified in the user's directory entry.

Operands

? specifies that the existing CRYPTO statement is to be queried.

DELETE

specifies that the existing CRYPTO statement is to be deleted from the user directory.

DOMAIN *domains*

specifies a list of 1 to 16 domains the virtual machine may use. Domains are the integer numbers 0 through 15 inclusive without duplication. If CAM (Crypto-Asynchronous-Message) queues are available, the virtual machine may also use the CAM queues corresponding to the domains specified.

CSU *

tells CP to assign this virtual machine's virtual crypto facilities to real crypto units. On CMOS cryptographic processors (supported by VM/ESA 2.4.0 and later), this also tells CP to allow this virtual machine to use both Crypto Asynchronous Processor (CAP) 1 and CAP 0. The default is CSU (*), if you do not specify CSU.

CSU 0

tells CP to

- Assign this virtual machine's virtual crypto facility to real crypto unit 0.
- Allow this virtual machine to use CAP 0.

CSU 1

tells CP to assign this virtual machine's virtual crypto facility to real crypto unit 1. Also tells CP to allow this virtual machine to use CAP 1.

KEYENTRY

tells CP that this virtual machine is authorized to enter keys through the key-entry unit on the cryptographic facility provided on Bipolar processors.

MODIFY

tells CP that this virtual machine can issue the PKSC (Public Key Security Control) Modify instructions.

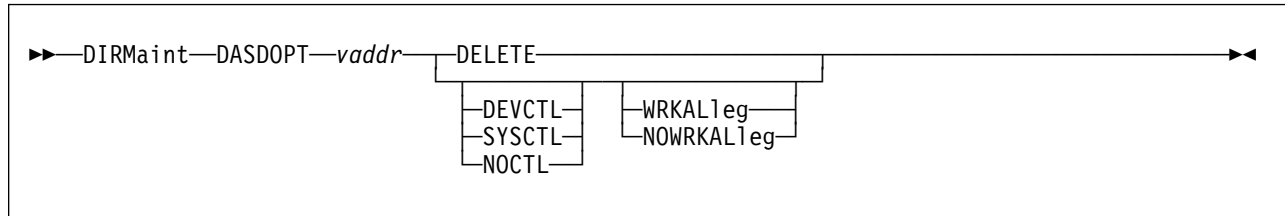
SPECIAL

tells CP that this virtual machine can use special security mode.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the CRYPTO directory statement, see *z/VM: Planning and Administration*.
3. The CSU, KEYENTRY, MODIFY, and SPECIAL operands may be specified in any order.

DASDOPT



VM Systems: VM/ESA (ESA Feature)

Purpose

The DASDOPT operand of the DIRMAINT command can be used to add, change, or delete an existing DASDOPT statement.

Operands

vaddr

is the virtual address of the MDISK, DEDICATE, or LINK statement that the new or changed DASDOPT statement should apply.

DELETE

means that the specified DASDOPT directory control statement should be deleted.

DEVCTL

means that CCWs that have an effect on system resources and functions directly related to the device will be accepted.

SYSCTL

means that CCWs that can have a direct global effect on subsystem resources and functions, not just those pertaining to the device, will be accepted.

NOCTL

means that CCWs that exercise or control subsystem resources or functions will not be accepted.

WRKALLEG

NOWRKALLEG

specifies that working allegiance is active or is not active on the minidisk.

Usage Notes

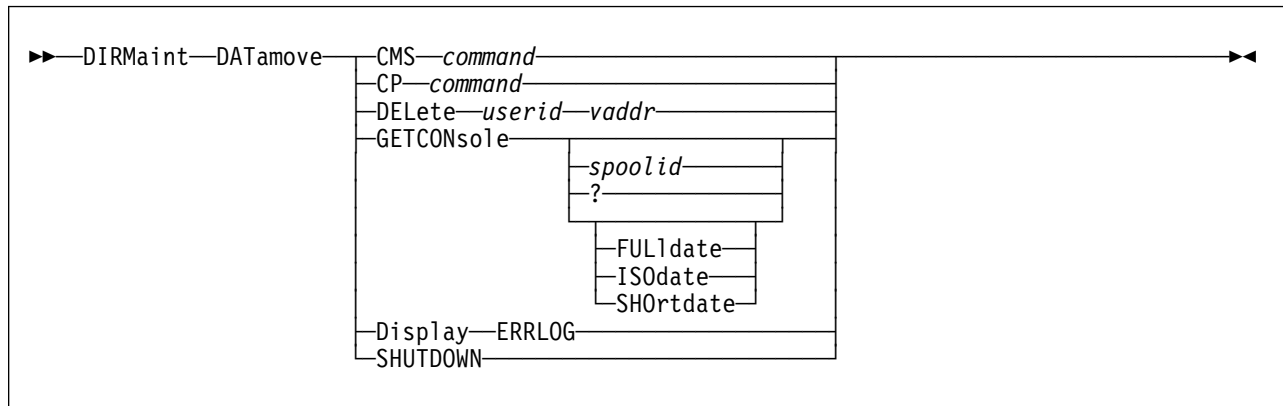
1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the DASDOPT directory statement see *z/VM: Planning and Administration*.
3. A DASDOPT statement is a logical extension to a DEDICATE, MDISK, or LINK statement in the directory. It is used to specify the extent of control a virtual machine has over the caching function of a full-volume minidisk or a

dedicated device that resides on a cache control unit. DASDOPT statements should only be used with minidisks that are full-volume minidisks.

Note: DirMaint does not enforce the full-volume minidisk requirement.

4. Do not place comments or blanks between an MDISK, DEDICATE, or LINK statement and the associated DASDOPT statement if it is manually added to a directory definition.

DATAMOVE



VM Systems: All

Purpose

The DATAMOVE operand of the DIRMAINT command is used to cause DirMaint to direct the DATAMOVE virtual machine to perform the specified function.

Operands

CP *command*

CMS *command*

tells the DATAMOVE service machine to execute the specified CP or CMS command.

DELe *userid vaddr*

specifies that the DATAMOVE virtual machine is to delete the entry specified as *userid vaddr* from the retry queue file.

Display **ERRLOG**

specifies that a copy of the DATAMOVE machine's retry queue file is to be sent to the requestor's virtual reader as DVHDMX ERRLOG.

GETCON *sole spoolid*

specifies that a copy of the identified console spool file is to be sent to the command issuer. If *spoolid* is omitted, the currently active console spool file is closed and a copy is sent to the command issuer.

Note: Specifying the spool ID of the currently active console spool file will result in an error. To obtain a copy of the currently active console spool file, omit the *spoolid* parameter.

GETCON **sole ?**

specifies that a list of the available console spool files is displayed to the command issuer.

FULldate

Uses the FULLDATE option on the CP QUERY PRT command. This is valid for VM/ESA 2.2.0 and follow-on releases.

ISOdate

Uses the ISODATE option on the CP QUERY PRT command. This is valid for VM/ESA 2.2.0 and follow-on releases.

SHORtdate

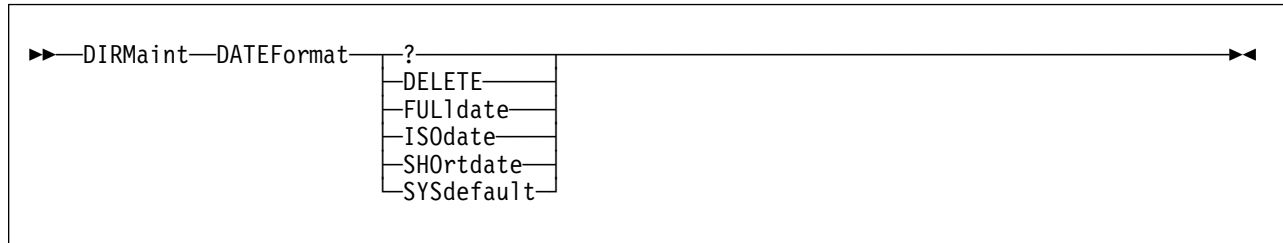
Uses the SHORTDATE option on the CP QUERY PRT command. This valid for VM/ESA 2.2.0 and follow-on releases.

SHUTDOWN

specifies that the DATAMOVE virtual machine session is to be terminated.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The DATAMOVE SHUTDOWN operand is queued on the DATAMOVE virtual machine command queue and will be processed in sequence. Any commands remaining on the queue after the SHUTDOWN operand will not be processed.
3. The DVHDMX ERRLOG file is a copy of the DATAMOVE machine's retry queue file created when requested by the DIRM DATAMOVE DISPLAY ERRLOG command. The retry queue file contains a record of work to be retried because a CP LINK command failed to obtain access to a disk. The retry will take place the next time the object directory is updated. Other errors encountered while processing a FORMAT, COPY, or CLEAN request are "rolled back," with the user's original disk being returned to the user and no record of the failure in the retry queue on the DATAMOVE machine. The console spool file and optional TRANSLOG file contains this historical information. The DIRMAINT machine keeps a record of failed work units. This includes errors encountered in the DATAMOVE machine processing. This information can be displayed using the DIRM STATUS WORKUNIT FAILED command.
4. For more information about the FULLDATE, ISODATE, or SHORTDATE operands, use a HELP CP QUERY command, then press PF11 and select PRINTER; or see *z/VM: CP Command and Utility Reference*.

DATEFORM


VM Systems: VM/ESA 2.2.0 and later

Purpose

Use the DATEFORMAT operand of the DIRMAINT command to allow the general user to add, change, delete, or query the DATEFORMAT statement in a directory entry.

Operands

? issues a query against the existing DATEFORMAT statement.

DELETE

deletes an existing DATEFORMAT statement from the directory entry.

FULLdate

requests that dates be shown in FULL date, *dd/yyyy*, format.

ISOdate

requests that dates be shown in ISO date, *yyyy-mm-dd*, format.

SHORtdate

requests that dates be shown in SHORT date, *mm/dd/yy*, *mm/dd*, or *dd mmm yy* format depending on the particular command.

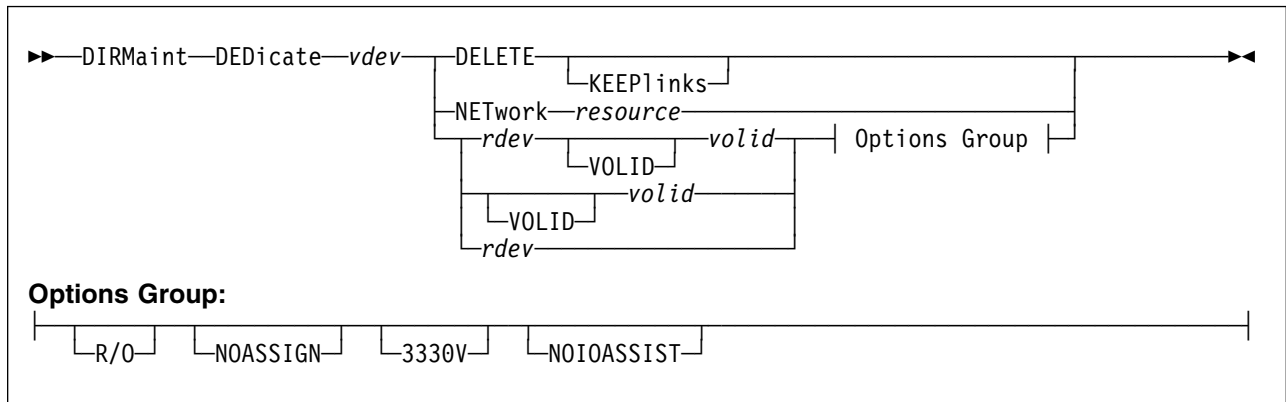
SYSdefault

requests that dates be shown in whichever format has been selected by the system administrator and specified with the SYSTEM_DATEFORMAT statement in the system configuration file, or by the system operator and specified with a CP SET DATEFORMAT SYSTEM command.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the DATEFORMAT directory statement, see *z/VM: Planning and Administration*.

DEDICATE



VM Systems: All

Purpose

The DEDICATE operand of the DIRMAINT command can be used to add or delete an existing DEDICATE statement.

Operands

vdev

is the virtual address associated with the DEDICATE statement.

DELETE

means that the specified DEDICATE directory control statement should be deleted.

KEEPlinks

specifies that the links to the deleted volume are to be kept. This allows for replacement of the deleted volume without having to reestablish the links.

NETWORK

is the keyword used if a remote 3270 information display system printer is to automatically attach to a virtual machine at logon time.

resource

is the four character resource ID of a remote device as specified in the DMKRIO.

rdev

is the real device number.

VOLID

specifies that the value following it is a volume label.

valid

is the volume label of a disk pack mounted on a real disk storage device. It must be a 1 to 6 character string. If *valid* is less than 5 characters, the VOLID keyword is required to distinguish it from an *rdev* address.

R/O

specifies that the virtual device is to be in read-only mode.

DEDICATE

NOASSIGN

indicates that the attach process should not issue an assign channel command for this user.

3330V

specifies that all interruptions, including cylinder faults and attentions received on the *rdev* are to be passed to the virtual machine in its *cuu*.

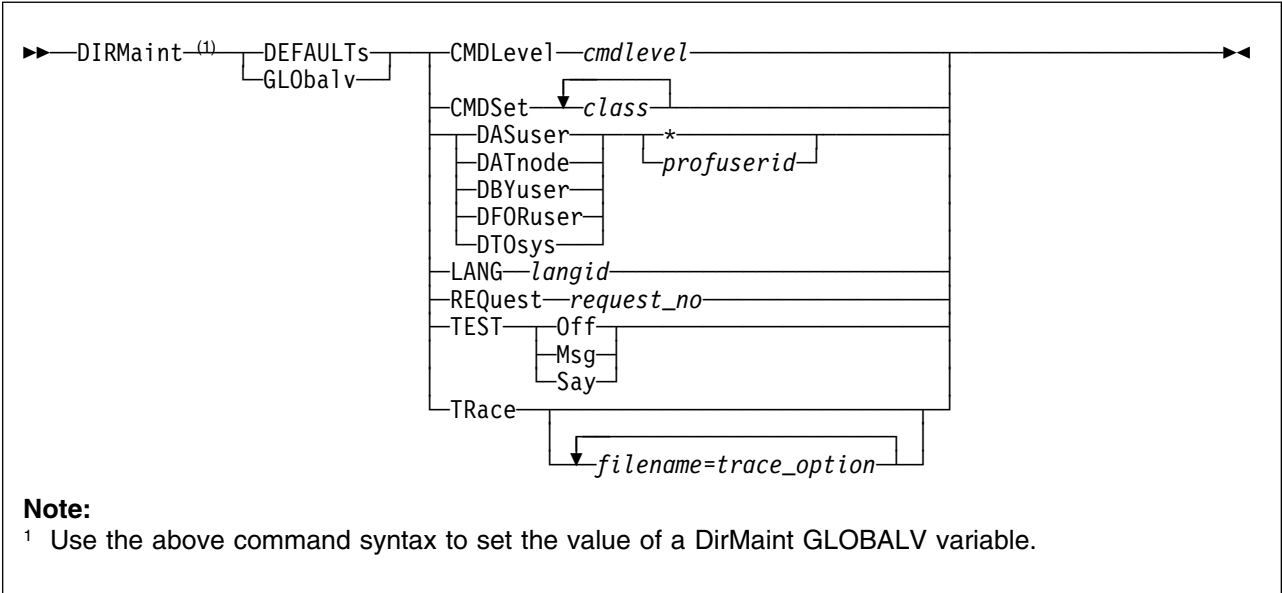
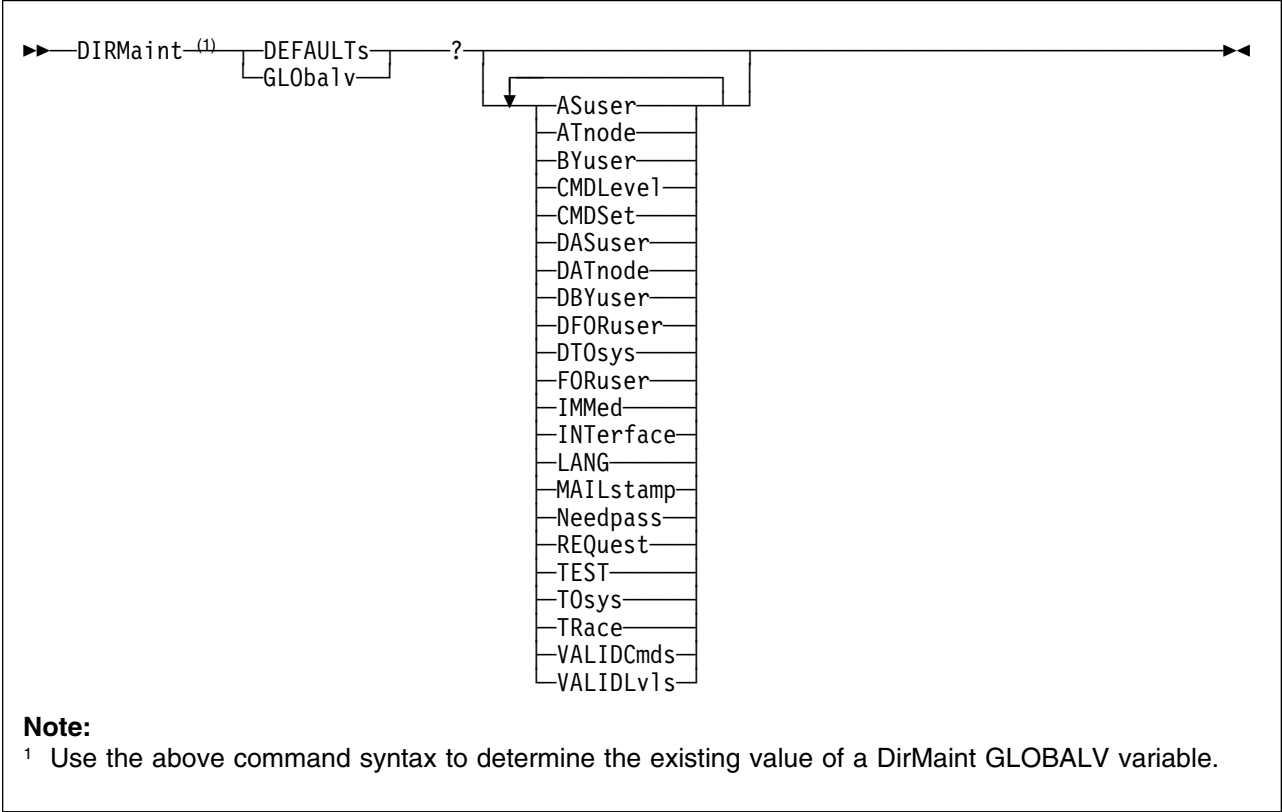
NOIOASSIST

indicates the device is not eligible for IOASSIST.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the DEDICATE directory statement see *z/VM: Planning and Administration*.
3. If more than one virtual machine has a dedicate statement for a given real device, only the first virtual machine to log on receives control of the device.
4. Note that all options are not available on all releases of VM/ESA. Consult the documentation for your specific release for a detailed description.

DEFAULTS, GLOBALV



VM Systems: All

DEFAULTS

Purpose

The DEFAULTS or GLOBALV operand of the DIRMAINT command is used to set or query various default values used within DirMaint processing.

Operands

where:

? Causes the current value of one or more variables to be displayed.

ASuser

Is set by the program from the ASUSER prefix keyword on the DIRM command (if specified), or from the DASUSER global variable (if previously set), or to an * by default.

ATnode

Is set by the program from the ATNODE prefix keyword on the DIRM command (if specified), or from the DATNODE global variable (if previously set), or to an * by default.

BYuser

Is set by the program from the BYUSER prefix keyword on the DIRM command (if specified), or from the DBYUSER global variable (if previously set), or to an asterisk (*) by default.

CMDLevel *cmdlevel*

May be set by the user to indicate the compatibility level desired for use with application programs. Valid *cmdlevel* values are 150A or 140A.

CMDSet *class(es)*

May be set by the user to indicate the command set the user believes is authorized for his or her use. This is expressed as a single string of 1 or more characters. The valid *class* values are tailorable by the installation. The IBM supplied default command classes are:

G for General Users
A for System Administrators
D for DASD Management Staff
H for the Help Desk Team
M for password monitoring commands
O for the System Operator
P for automated programs such as DFSMS/VM®
S for the System Programmer
Z for internal use between the DIRMAINT and DATAMOVE virtual machines

Up to 36 characters may be specified, in any order, with no spaces in between them.

The DirMaint Release 5 classes equate to the DirMaint Release 4 authorities as follows:

| DirMaint Release 5 classes | DirMaint Release 4 authority |
|---|---|
| G S | OWNER |
| G O | SYS_OPER |

| | |
|-----------|----------------|
| G A D H P | DIRM_STAFF |
| G H | DIRM_SUB_STAFF |
| G M | PWMON |

DASuser

Identifies the default value for the ASUSER prefix keyword on the DIRMAINT command. Valid values are an asterisk or a valid file name which may be used as a user ID.

DATnode

Identifies the default value for the ATNODE prefix keyword on the DIRM command. Valid values are an asterisk or a valid file name which may be used as a network node ID.

DBYuser

Identifies the default value for the BYUSER prefix keyword on the DIRM command. Valid values are an asterisk or a valid file name which may be used as a user ID.

DFORuser

Identifies the default value for the FORUSER prefix keyword on the DIRM command. Valid values are an asterisk or a valid file name which may be used as a user ID.

DTOsys

Identifies the default value for the TOSYS prefix keyword on the DIRM command. Valid values are an asterisk or a valid file name which may be used as a network node ID.

IMMed

Is set by the program based on the presence or absence of the IMMED keyword prefix on the DIRM command.

LANG *langid*

A valid *langid* consists of 5 characters or less, that are valid in a CMS file name, and is explicitly defined on one or more *langid* USER MSGS entries in the CONFIG* DATADVH file(s).

Also a *langid* of *1SAPI* may be used to specify the *application programming interface* language, where all replies are:

```
REQUEST= nnnn RTN=DVHXXX MSGNO=1234 FMT=01
SUBS= VAR1 VAR2
```

MAILstamp

Displays the last time the mail file was sent to you.

Needpass

Is set by the NEEDPASS command to indicate whether password prompting is done for most DirMaint commands (YES) or not (NO).

REQuest *request_no*

Specifies an identifying number to be associated with a transaction sent to the DIRMAINT server. Used by programs using LANG 1SAPI to associate the transactions with the returned messages.

TEST

Determines whether or not a DIRM command results in a transaction to the DIRMAINT service machine (TEST OFF), or is displayed on the invoker's

DEFAULTS

console as a CP message (TEST MSG), or is displayed on the invoker's console using a CMS REXX SAY statement (TEST SAY). Valid values are OFF, MSG, or SAY.

TOSys

Is set by the program from the TOSYS prefix keyword on the DIRM command (if specified), or from the DTOSYS global variable (if previously set), or to an * by default.

TRace

Is used for problem diagnosis to activate REXX tracing for the listed routines, and specify the degree of detail desired in the trace. Each entry must be in the form of a file name, followed by an equal sign, followed by a REXX trace option (All, Commands, Error, Failure, Intermediates, Labels, Normal, Off, Results, Scan); with no intervening blanks.

VALIDCmds

Is set by the program to identify the valid values for CMDLEVELS. The valid values are 150A and 140A.

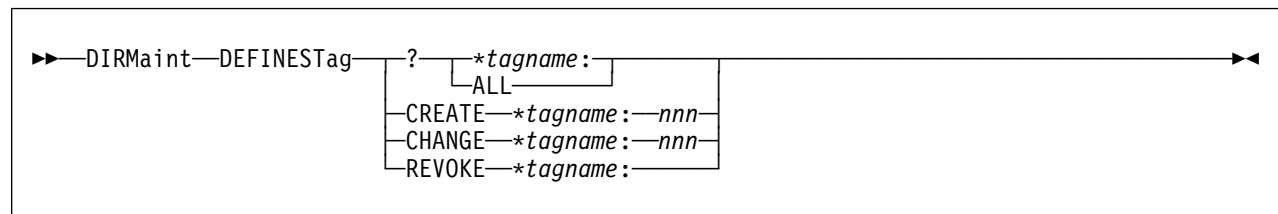
VALIDLvls

Is set by the program to identify the valid values for the interface level. The current value is 199501.

Usage Notes

1. DirMaint prefix operands do apply to this command. They may be accepted but will be ignored.
2. The DEFAULTS and GLOBALV commands are completely handled within the virtual machine of the command issuer. No transaction is sent to the DIRMAINT service machine, therefore no password is required or requested for authentication, even if you have set NEEDPASS to YES.
3. For more information on altering the NEEDPASS setting, see "NEEDPASS" on page 156.
4. The DEFAULTS and GLOBALV commands are not valid within a BATCH command file.
5. All global variables are stored in the DVH15 variable pool.

DEFINESTAG



VM Systems: All

Purpose

The DEFINESTAG operand of the DIRMAINT command is used to manipulate user defined tagged comments. An installation can define local tags that can be stored in the CP directory and manipulated by DirMaint. This may be useful for information normally placed into comments, such as department information. This command creates the required definitions within the DIRMAINT machine. It is not used to assign data to a local tag, it is only used to create and manipulate a local tag.

Operands

? requests that the current setting of the specified tag (or all tags, using the ALL keyword) be returned in a message.

***tagname:**

is a user defined tag name. The following rules apply:

- It must start with an "*" and end with a "."
- The body of the tag (between the "*" and the ".") must be 10 characters or less.
- The body of the tag is restricted to alphanumeric characters.

ALL

requests that all tags be reported in the query.

CREATE

attempts to create a new tag.

CHANGE

attempts to alter an existing tag definition's sort sequence.

nnn

specifies the sort order of the created tag in the range of 000 to 999.

REVOKE

requests that the specified tag be removed from DirMaint's internal tables. The tag can no longer be set or queried by users.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The sort value (*nnn*) used with CREATE and CHANGE assigns a unique sort order to each user defined tag. Internally, this sort order is *8nnn* . The *nnn* is used to assign a relative sorting order in regards to user defined tags only. It does not affect the sorting order of CP directory statements.
3. Currently the number of user defined tags is limited to 1000 (using sort orders 000 to 999).
4. When using the CREATE keyword, the tag being created must not already exist and the provided sort order code must not be in use by any other tag.
5. When using the CHANGE keyword, the tag name being changed must already exist and the new sort order must not already be in use.
6. When using the REVOKE keyword, DirMaint makes sure that the specified tag is not being used in any directory. If it is, the request will fail. If you wish to disable this tag and do not wish to remove it from all user directories, the file containing the tag definition information (LOCAL DATADVH) can be altered on the DIRMAINT server. It is located on the primary directory disk. Once an entry is removed from this file, the tags in the user directories no longer have special meaning to DirMaint and are treated like comments.
7. The widespread use of local tags has the ability of significantly expanding the size of the source directory. This should be kept in mind when defining local tags and authorizing users to use them.

DIRECT

▶▶—DIRMaint—DIRECT————▶▶

VM Systems: All

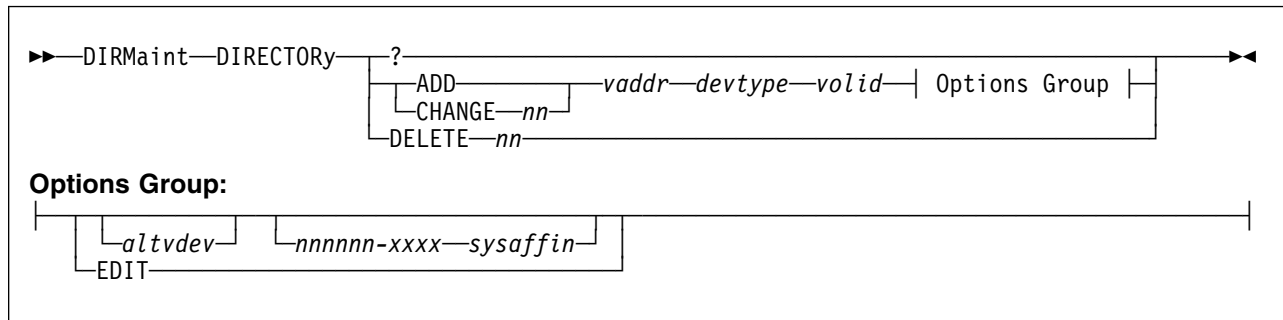
Purpose

The DIRECT operand of the DIRMAINT command is used to place the current directory structure online.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the DIRECT/DIRECTXA command, see *z/VM: CP Command and Utility Reference*.
3. DIRECT/DIRECTXA (depending on system level) is used to place the directory online.
4. Any errors reported by the VM directory program are reflected in the DirMaint messages.
5. Should an error result during the operation of this command (issued from the CP DIRECT/DIRECTXA module), DirMaint will disable source directory updates automatically. The source directory can be enabled for updates using the DIRM ENABLE command.

DIRECTORY



VM Systems: All

Purpose

The `DIRECTORY` operand of the `DIRMAINT` command is used to add, change, or delete the `DIRECTORY` directory control statement. The `DIRECTORY` statement defines to CP the device on which space has been allocated for the directory.

Operands

? returns the settings of the current `DIRECTORY` directory control statement specifications.

ADD

specifies that a `DIRECTORY` directory control statement containing the specified information is to be added to the system directory.

CHANGE *nn*

requests that the provided information replace the specified existing directory statement.

vaddr

specifies the virtual device number of the device that is used to contain the object directory.

devtype

specifies the device type. Valid device types are:

- 3350 9332
- 3370 9335
- 3375 9336
- 3380 9345
- 3390 FB-512

valid

specifies the volume label of the directory volume. The variable *valid* is a 1 to 6 character alphanumeric string.

altvdev

specifies an alternative virtual device number, on which to write the directory if the primary virtual device number is unavailable.

EDIT

defines a special work volume to be used by the DIRECT command when it is entered with the EDIT option. You can only specify the EDIT option on one DIRECTORY statement, and it must be the last of the set of DIRECTORY statements. DIRECTXA validates the syntax of this statement, but ignores its contents.

nnnnnn-xxxx

specifies the processor ID of the system to which the DIRECTORY statement applies. If the same *nodeid* is used for all CPUs in an *n*-way processor, the first character of the serial number may be specified as an asterisk, for example **nnnnn-xxxx*

sysaffin

specifies a 1 to 8 character alphanumeric string that identifies the system whose object directory is affected by the SYSAFFIN statements.

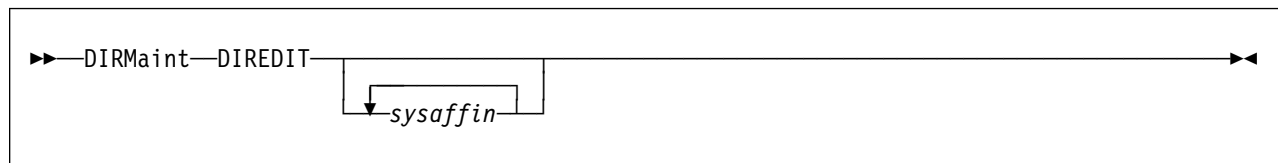
DELETE *nn*

requests that the specified existing directory statement (the *nn*'th directory statement) be deleted.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the DIRECTORY directory statement, see *z/VM: Planning and Administration*.

DIREEDIT



VM Systems: All

Purpose

The DIREEDIT operand of the DIRMAINT command is used to syntactically check the current directory structure through the use of the EDIT feature of the VM directory program.

Operands

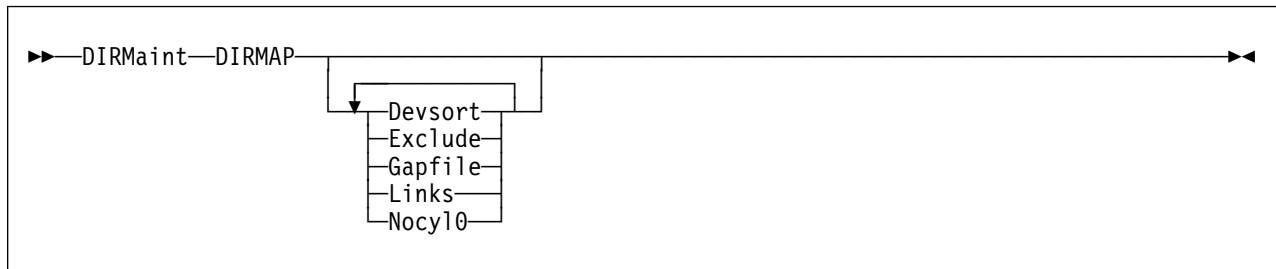
sysaffin

specifies that the edit operation should pass the *sysaffin* value to the DIRECT/DIRECTXA module.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The directory is not placed online, it is only checked for syntax.
3. DIRECT/DIRECTXA (depending on system level) is used to edit check the directory structure.
4. For more information on the DIRECT/DIRECTXA command, see *z/VM: CP Command and Utility Reference*.
5. Any errors reported by the VM directory program are reflected in the DirMaint messages.
6. Should an error result during the operation of this command (issued from the CP DIRECT/DIRECTXA module), DirMaint will disable source directory updates automatically. The source directory can be enabled for updates using the DIRM ENABLE command.

DIRMAP



VM Systems: All

Purpose

The DIRMAP operand of the DIRMAINT command is used to provide a report detailing the current DASD utilization on the system. Many options supported in the VM/ESA CMS Utilities Feature DIRMAP command are also provided within this DIRMAINT command.

Operands

Devsort

requests that the report is to be sorted by device type followed by volume label. The default is to sort by volume label.

Exclude

specifies that EXCLUDED extents are not to be considered when generating reports.

Gapfile

Generate a file listing gaps in addition to a report detailing the current DASD utilization.

Note: If you use the GAPFILE option without using the EXCLUDE option and excluded full volume overlays are defined on your system no gap data will be returned. Use the EXCLUDE option to eliminate the excluded full volume overlays from consideration when building the gap data. The gapfile is listed in the following format: (note that all fields are blank delimited and not bound to a specific column range)

| Field | Use |
|-------|-----------------------------------|
| 1 | valid of disk |
| 2 | device type of disk. |
| 3 | start cylinder/block of gap |
| 4 | end cylinder/block of gap |
| 5 | number of cylinders/blocks in gap |

Links

requests that a LINKMAP be produced. This file contains a map of all LINKs to MDISKs on the processed volumes. The default is to only generate a DISKMAP.

DIRMAP

Nocyl0

Forces records that are destined for the GAPFILE to be recomputed to use cylinder 1 as a start to prevent a gap being identified as starting on cylinder 0. This option is only meaningful to GAPFILE output. The Mdisk-map shows cylinder 0 as a gap (if it is not used) regardless of this option.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Since DIRMAINT only maintains active volume control files for non-full volume minidisks, the DASD map reports generated by this command do not include any full volume minidisk entries.

DISABLE



►►—DIRMaint—DISable—◄◄

VM Systems: All

Purpose

The DISABLE operand of the DIRMAINT command causes incoming directory updates to be rejected.

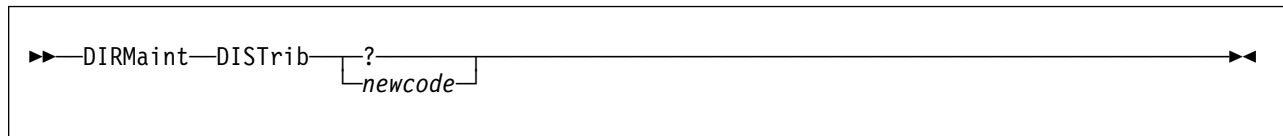
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Use this operand to temporarily suspend updates to the source directory. When this operand is used, read-only requests will still be processed.
3. The disable function of this operand is automatically invoked if the DIRECT/DIRECTXA program returns an error when attempting to place the directory online.

Note: This only occurs as the result of a DIRM DIRECT command and indicates a problem with the directory structure.

4. Normal operation can be resumed by invoking the DIRMAINT ENABLE operand.

DISTRIB



VM Systems: All

Purpose

The DISTRIB operand of the DIRMAINT command changes the distribution code that is printed or punched with your user ID in the separator for spooled output. The distribution code, if defined, exists as a parameter on the ACCOUNT control statement in your directory entry.

Operands

? allows you to query your current distribution code.

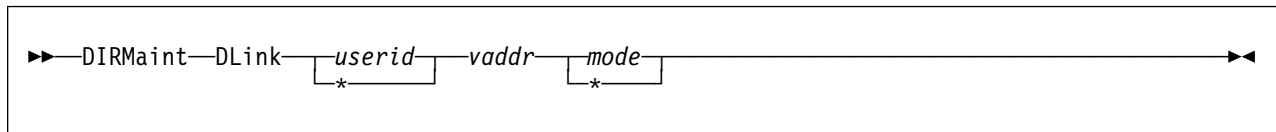
newcode

is the new distribution code. This is the value that appears below your user ID on printed output.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Because of the query capability for this operand, a distribution code of “?” can not be installed in a user's directory definition using the DISTRIB operand. If such a distribution code is desired, it must be set using the GET and REPLACE operands.
3. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

DLINK



VM Systems: All

Purpose

The DLINK operand of the DIRMAINT command removes LINK statements.

DLINK does not physically alter the source directory. It creates a batch file containing one or more

```
FOR userid LINK * addr addr DELETE
```

statements. The batched commands run as low priority background work that removes the LINK statements from the linking user IDs' directory definitions.

Operands

userid

specifies the name of the user whose link to this minidisk is to be dropped. All user links to this minidisk can be dropped by specifying an asterisk (*) instead of a specific user ID.

vaddr

specifies the virtual address of the minidisk from which links are to be dropped.

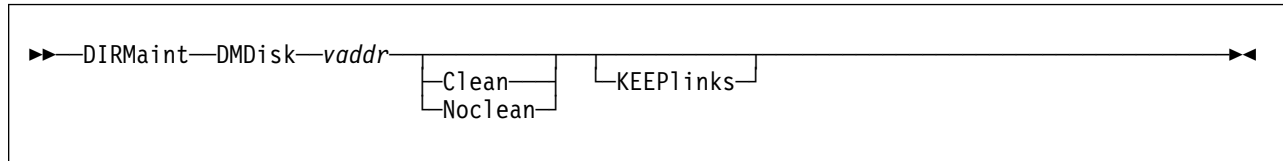
mode

specifies the type of link to be dropped. All links regardless of the LINK mode can be dropped by specifying an asterisk (*) instead of a specific MODE. The file modes that can be used are: R, W, M, or *.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. To determine the links that are in effect to your minidisks, issue
DIRM REVIEW
The returned output will contain a list of all links to your minidisks, identifying the linking user ID and their link mode.
3. The ANYlink and DIRMlink operands are no longer a part of the command syntax. If specified, they are tolerated for compatibility with earlier releases of DirMaint.

DMDISK



VM Systems: All

Purpose

The DMDISK operand of the DIRMAINT command is used to delete a minidisk from a user directory entry.

Operands

vaddr

specifies the virtual device address of the minidisk.

Clean

specifies that the released minidisk extent is to be transferred to the user defined as a DATAMOVE machine in the CONFIG* DATADVH file for formatting, even if the DISK_CLEANUP= YES option is not in effect (see Usage Note 3).

Noclean

specifies that the released minidisk extent is to be made available for reassignment immediately without being cleaned, even if the DISK_CLEANUP= YES option is in effect (see Usage Note 3).

KEEPlinks

may be specified to suppress the removal of links to the minidisk being deleted. The default is to remove any links to a minidisk being deleted by the DMDISK operand.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The extent currently occupied by this minidisk will become available for reallocation immediately, unless the installation wants obsolete extents to be formatted before reassignment.
3. If the installation wants released minidisks to be formatted before reassignment, then two CONFIG* DATADVH file statements are consulted.
 - DISK_CLEANUP=
 - CYL0_BLK0_CLEANUP=

DISK_CLEANUP determines (in general) whether or not minidisk cleans should take place. CYL0_BLK0_CLEANUP is consulted if the extent to be cleaned is using cylinder or block 0. In addition, minidisks will not be cleaned if they form an overlap with another uninvolved extent.

If cleaning is required, the minidisk will be transferred to the user defined as a DATAMOVE machine in the CONFIG* DATADVH file. A table entry is made denoting that the minidisk is to be formatted. The old extent will not be available for reassignment until DIRMAINT is notified that the extent has been formatted.

This processing can be controlled on a transaction basis. If CLEANUP is the normal mode of operation, it can be bypassed by the NOCLEAN option. If CLEANUP is not in effect, it can be forced for a given transaction by including the CLEAN option. This overrides the action dictated by the DISK_CLEANUP= and CYL0_BLK0_CLEANUP= configuration file entries.

The CLEAN option cannot be used if the subject *userid* and the DATAMOVE *userid* are the same.

4. DMDISK processing will schedule low priority work to remove all links to the minidisk being deleted unless the KEEPLINKS option is specified.
5. If one of the DATAMOVE virtual machine's minidisks is specified to be cleaned the request will be rejected.
6. Be aware that the minidisk being deleted may be physically linked by other users at the time of its deletion. The DATAMOVE virtual machine postpones the formatting of any disk until it is free of links. It is the installation's responsibility to ensure that these physical CP links are removed. Failure to remove the CP links may result in significant delay before the extents are available for reuse.
7. A DEVNO MDISK may be deleted with the DMDISK command.

Note: DISK_CLEANUP= YES entry in the CONFIG* DATADVH file and the CLEAN option on the DMDISK command are both ignored for DEVNO MDISKs.

DEVNO MDISKs will not be formatted by DATAMOVE. If there is residual data that must be protected, it is the responsibility of the owning user to format the disk or otherwise erase the data before requesting the administrator to delete the minidisk or purge the user ID.

8. A T-DISK or V-DISK may be deleted using DMDISK. The space for these may or may not be cleaned by the system before reuse. The DISK_CLEANUP= YES entry and the CLEAN option on the DMDISK command are both ignored for these disks.

DROPBY



VM Systems: All

Purpose

The DROPBY operand of the DIRMAINT command is used to revoke the authority given by DIRM AUTHBY to utilize the BYuser prefix operand.

Operands

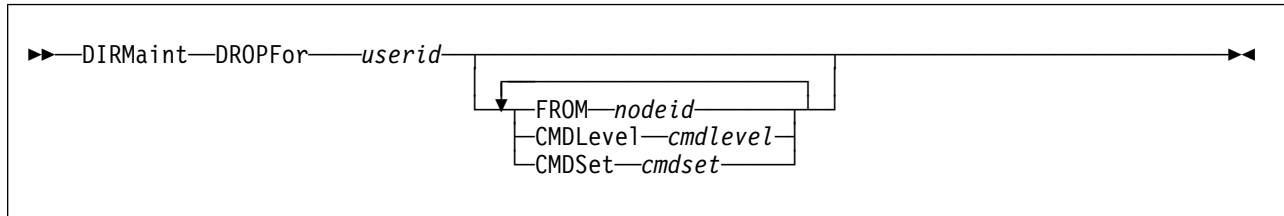
userid

a list of one or more user IDs whose authorization is revoked.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

DROFFOR



VM Systems: All

Purpose

The DROFFOR operand of the DIRMAINT command is used to revoke the authority granted by the AUTHFOR command to issue a command on behalf of another user.

Operands

userid

a user ID whose authorization is revoked.

FROM *nodeid*

specifies the node ID of the user whose authorization is being dropped.

CMDLevel *cmdlevel*

specifies the command level being dropped.

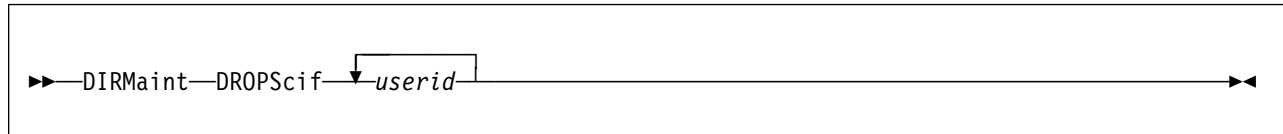
CMDSet *cmdset*

specifies the command set being dropped.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

DROPCIF



VM Systems: All

Purpose

The DROPCIF operand of the DIRMAINT command is used to revoke the authority granted by a previous AUTHSCIF command to utilize another user as a secondary console.

Operands

userid

is a list of one or more users having their authority revoked.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The use of SECUSER, AUTHSCIF, DSECUSER and DROPCIF commands are closely related. The *userid* must have granted the issuing user authority through the AUTHSCIF command to use them as a secondary console. An error message will result if this has not been done. For example, if the *userid* WATCHER chooses to become the secondary console of the disconnected user SERVER, this sequence of commands is needed:

- a. WATCHER issues DIRM AUTHSCIF SERVER
- b. SERVER issues DIRM SECUSER WATCHER

When the secondary user facility is no longer needed, one of two commands would be needed:

- a. WATCHER issues DIRM DSECUSER SERVER
- b. SERVER issues DIRM SECUSER OFF

If SERVER is not going to use WATCHER as a SCIF user again, the general authority can be cancelled if WATCHER issues:

DIRM DROPCIF SERVER

The design allows the SCIF target *userid* (WATCHER) to have overall control of the usage of SCIF, in case of SERVER produced SCIF output which the target was not able to handle. SERVER has sole control of the time it is turned on, but WATCHER must authorize it first. If the SECUSER command is issued before the nominated *userid* has granted authority with the AUTHSCIF command, an error message will result.

DSECUSER

```
▶▶—DIRMaint—DSECuser—userid—————▶▶
```

VM Systems: All

Purpose

The DSECUSER operand of the DIRMAINT command is used to request that a user, who is currently using the command invoker as a secondary console, drop the entry from their CONSOLE statement.

Operands

userid

the user who is currently pointing at the command issuer as a secondary console.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The use of SECUSER, AUTHSCIF, DSECUSER and DROPSCIF commands are closely related. The *userid* must have granted the issuing user authority through the AUTHSCIF command to use them as a secondary console. An error message will result if this has not been done. For example, if the *userid* WATCHER chooses to become the secondary console of the disconnected user SERVER, this sequence of commands is needed:
 - a. WATCHER issues DIRM AUTHSCIF SERVER
 - b. SERVER issues DIRM SECUSER WATCHER

When the secondary user facility is no longer needed, one of two commands would be needed:

- a. WATCHER issues DIRM DSECUSER SERVER
- b. SERVER issues DIRM SECUSER OFF

If SERVER is not going to use WATCHER as a SCIF user again, the general authority can be cancelled if WATCHER issues:

```
DIRM DROPSCIF SERVER
```

The design allows the SCIF target *userid* (WATCHER) to have overall control of the usage of SCIF, in case of SERVER produced SCIF output which the target was not able to handle. SERVER has sole control of the time it is turned on, but WATCHER must authorize it first. If the SECUSER command is issued before the nominated *userid* has granted authority with the AUTHSCIF command, an error message will result.

DUMP

DUMP

▶▶—DIRMaint—DUMP—▶▶

VM Systems: All

Purpose

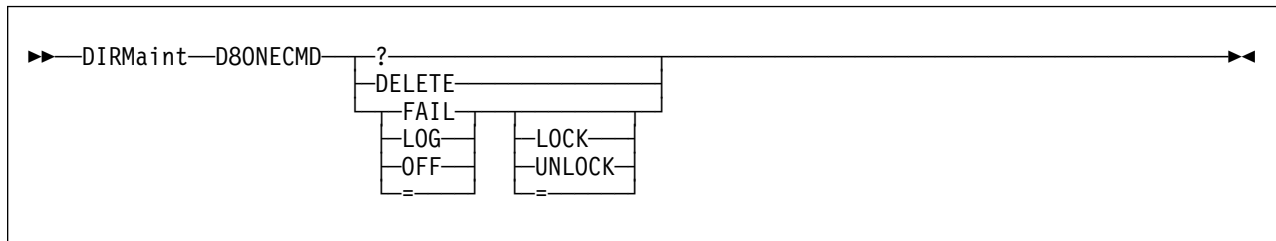
The DUMP operand of the DIRMAINT command is used to request a variable dump of the DIRMAINT virtual machine.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. A storage dump is not taken for Release 5. Instead several values are queried and returned to the issuer of the command as a file. The queries that are issued are:

```
QUERY TIME
IDENTIFY
QUERY FILES *
QUERY USER
QUERY SECUSER
QUERY CPLEVEL
QUERY CPU ID
QUERY V STORAGE
QUERY SET
QUERY CON
QUERY UR
QUERY PF
QUERY CMSREL
QUERY CMSLEVEL
QUERY EXECMAP
QUERY NUCXMAP
GLOBALV LIST
```


D8ONECMD



VM Systems: VM/ESA (ESA Feature)

Purpose

The D8ONECMD operand of the DIRMAINT command allows the user to query, alter, or delete the current D8ONECMD directory statement.

Operands

? allows you to query your current D8ONECMD settings.

DELETE

allows you to delete the existing D8ONECMD statement.

FAIL

tells CP to prevent this virtual machine from issuing multiple commands with DIAGNOSE X'08', the first command is processed, subsequent commands are rejected.

LOG

tells CP that this virtual machine can issue multiple commands via DIAGNOSE X'08', logging is performed.

OFF

tells CP that this virtual machine can issue multiple commands via DIAGNOSE X'08', logging is not performed.

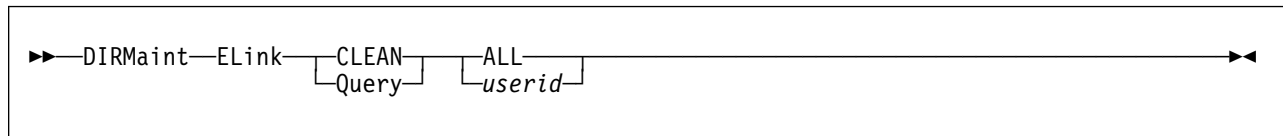
LOCK

UNLOCK

tells CP to lock or unlock the D8ONECMD setting for a specific user entry.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the D8ONECMD directory statement see *z/VM: Planning and Administration*.

ELINK


VM Systems: All

Purpose

The ELINK operand of the DIRMAINT command can be used by authorized users to:

- Restore a user's authorization to issue the LINK operand
- Perform maintenance on the LINK ATTEMPTS file
- Report status of all or any entries in the LINK ATTEMPTS file.

Operands

CLEAN *userid*

specifies the user ID whose DIRM LINK privileges are being restored. This is accomplished by removal of the user's record in the DVHLINK ATTEMPTS file.

CLEAN ALL

specifies that each record in the DVHLINK ATTEMPTS file is inspected. A record is only removed when the corresponding user has not issued an invalid DIRM LINK command in *nnn* days. The time period, *nnn* days is specified as the reset value in the LNKLMT statement in the CONFIG* DATADVH file(s).

Query *userid*

gives the status of a specified user ID in the DVHLINK ATTEMPTS file.

Query ALL

gives the status of all user IDs in the DVHLINK ATTEMPTS file.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The default value of LNKLMT= is 7 if not specified in the CONFIG* DATADVH file.
3. The DIRM ELINK CLEAN ALL command is used internally. It is performed automatically as scheduled in the DIRMAINT DATADVH wakeup times file. This daily cleanup activity should be sufficient for maintenance of the DVHLINK ATTEMPTS file.
4. Global maintenance of the DVHLINK ATTEMPTS file can be accomplished by use of the DIRM SEND and DIRM FILE operands.

When global maintenance is performed, the privileged user should be aware of the format of the *date* field in the DVHLINK ATTEMPTS file. The value is not in *mm/dd/yy form*. It is expressed in the ISO standard date format *yyyymmdd*.

5. The

LNKLMT= threshold reset_days

statement is a configuration statement that takes two parameters. The *threshold* defines the number of unsuccessful link attempts that are tolerated before locking. If missing or set to 0 this feature is disabled. The *reset_days* defines the number of days the user will be prohibited from issuing LINK commands. If missing, it defaults to 7 days.

ENABLE

ENABLE

▶▶—DIRMaint—ENable————▶▶

VM Systems: All

Purpose

The ENABLE operand of the DIRMAINT command is used to cancel a previous DISABLE operand, and allow incoming source update transactions to be accepted.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. This operand causes the DISABLE CONTROL file to be erased from the primary and secondary (if required) directory disks. The existence of this file disables updates to the directory.

EXECDROP

▶▶—DirMaint—EXECDrop—▶▶

VM Systems: All

Purpose

The EXECDROP command is used to undo the effects of an EXECLOAD command. It removes DirMaint processing routines from user resident storage. It also releases (and detaches) the minidisk or SFS directory on which the DirMaint interface files reside.

Usage Notes

1. DirMaint prefix operands do apply to this command. They may be accepted but will be ignored.
2. The EXECDROP command is completely handled within the virtual machine of the command issuer. No transaction is sent to the DIRMAINT service machine, therefore no password is required or requested for authentication, even if you have NEEDPASS set to YES.
3. A DIRM EXECLOAD should be preceded by a DIRM EXECDROP. This allows optimization of storage usage and ensures the interface disk or directory is re-accessed.
4. The EXECDROP command is not valid within a BATCH command file.
5. The files to be dropped are listed on the LOADABLE_USER_FILE= entries in the CONFIG* DATADVH file(s).

EXECLOAD

▶▶—DirMaint—EXECLoad—————▶▶

VM Systems: All

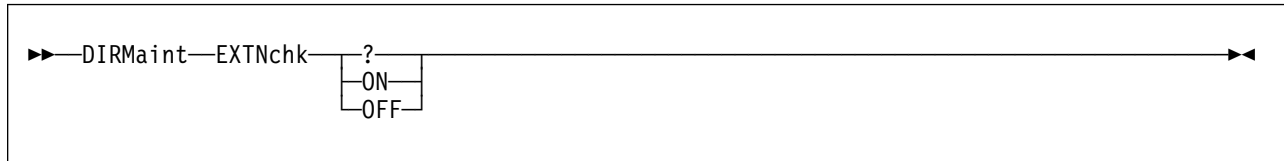
Purpose

The EXECLOAD command is used to improve performance for those users who frequently issue DirMaint commands, such as the administration staff. It links and accesses the minidisk or SFS directory on which the DirMaint interface files reside, and loads the executable code (EXECs, XEDITs, MODULEs) into storage. The effects of EXECLOAD last until reversed by use of a DIRM EXECDROP command, you re-IPL CMS, or logoff.

Usage Notes

1. DirMaint prefix operands do apply to this command. They may be accepted but will be ignored.
2. The EXECLOAD command is completely handled within the virtual machine of the command issuer. No transaction is sent to the DIRMAINT service machine, therefore no password is required or requested for authentication, even if you have NEEDPASS set to YES.
3. A DIRM EXECLOAD should be preceded by a DIRM EXECDROP. This allows optimization of storage usage and ensures the interface disk or directory is re-accessed.
4. The EXECLOAD command is not valid within a BATCH command file.
5. The files to be loaded are listed on the LOADABLE_USER_FILE= entries in the CONFIG* DATADVH file(s).

EXTNCHK



VM Systems: All

Purpose

The EXTNCHK operand of the DIRMAINT command is used to control MDISK extent validation.

Note

Use extreme care when extent checking is not active. Data corruption can occur if MDISK extents are allowed to overlap.

Operands

? requests that the current extent check value be returned.

ON

specifies that when a new minidisk is being allocated manually and a specific extent is being requested, the DIRMAINT virtual machine verifies that the extent is really available.

Note: This is the default condition. It is not necessary to set this parameter ON unless your configuration file indicates OFF or you have previously issued this operand with the OFF value.

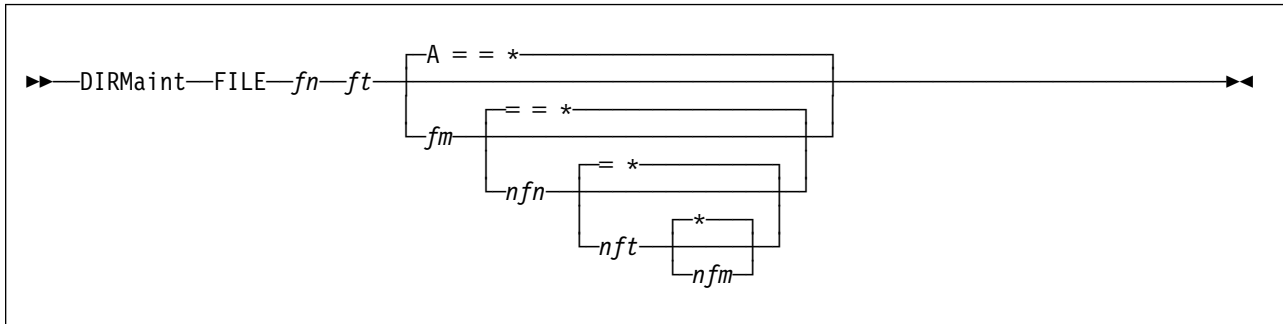
OFF

specifies that the validation of manually-allocated extents is inhibited. Use this operand if you deliberately want to allocate an overlapping minidisk.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The setting of EXTNCHK will be in effect for minidisks being added using explicit extent information.
3. The various automatic allocation methods provided by DirMaint can not be used to create overlapping minidisks. This setting is not consulted by these routines.

FILE



VM Systems: All

Purpose

The FILE operand of the DIRMAINT command is used to install or replace a file on one of DirMaint's minidisks.

Operands

fn ft fm

is the file identification and location on the invoker's minidisk. File mode defaults to A if no additional operands are needed.

nfn nft nfm

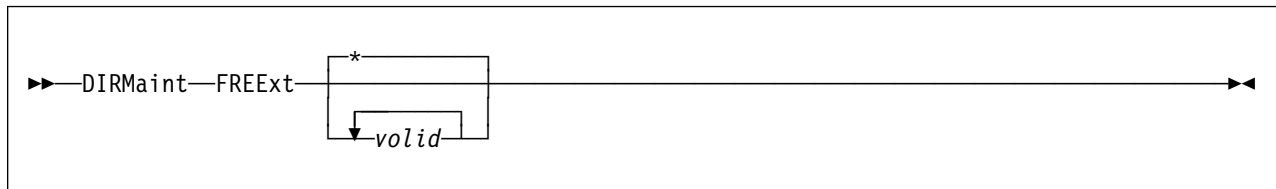
is the file identification and location as it is to be stored inside DirMaint. The *nfn* and *nft* operands may be omitted (and default to =) if the invoker's copy of the file already has the correct file identification and *nfm* is not specified.

The *nfm* operand is normally omitted. The file will be stored automatically on the correct DirMaint minidisk according to file name and file type.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. If the FILE operand is used to replace a part listed on a LOADABLE_SERV_FILE= or LOADABLE_DIRMAINT_FILE= record in the CONFIG* DATADVH file(s), then that operation should be followed with the RLDCODE operand to put the replaced file into operation. Otherwise, it is not placed in use until the next system restart.
3. This operand cannot be used to maintain DIRMAINT EXEC, DIRMSAPI EXEC SAMP or ACCESS DATADVH. They must reside on the CMS system disk, its extension disk, or the licensed program disk.

FREEXT



VM Systems: All

Purpose

The FREEXT operand of the DIRMAINT command is used to generate a listing of free extents on volume(s) used to contain minidisks.

Operands

valid

specifies the search domain.

If a 1 to 6 character volume label is specified, the listing will contain only allocatable extents on that volume.

If a partially-qualified volume label of 1 to 5 characters followed by an asterisk (*) is specified, the listing will include allocatable extents on all volumes beginning with the qualified *valid*. For example, if the total set of volumes containing minidisks are:

```
ABC
ABC12
ABC456
ABX445
VOL001
VSZ777
```

then specifying *valid* as ABC* will cause volumes ABC, ABC12 and ABC456 to be scanned. Specifying ABC would qualify only the discrete volume ABC. Specifying V* would cause volumes VOL001 and VSZ777 to be qualified.

If *valid* is omitted, or if an * is specified, then all volumes used to house minidisks will be included in the output.

Multiple discrete volumes (or multiple partially-qualified volumes) are also allowed.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The name of the file returned to the invoker of this command is dependant on the method used to invoke the command.

valid FREEXT

if the invocation was made with a specific volume specified.

FREEXT

`vo$ FREEXT`

if the invocation was made with a partially qualified valid specified.

`ALLVOLS FREEXT`

if the invocation was made with a * as the qualifier.

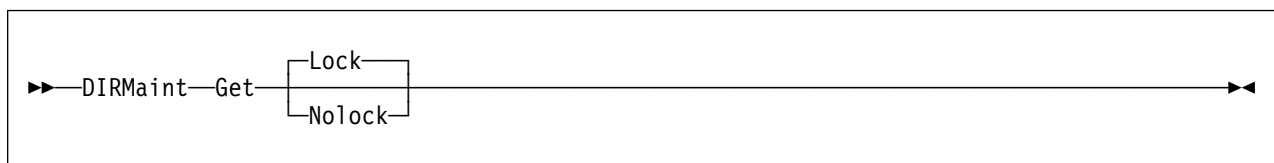
`MULTIVOL FREEXT`

if the invocation was made with more than one qualifier.

Note: When there are no free extents or used extents meeting the search criteria the corresponding file returned will contain only a header with no data.

3. If the command invocation results in multiple volumes being queried (either by specific volume specification or through the use of a partially qualified volume specification) each volume will appear in the report file sorted by volume ID.
4. If the volume architecture is CKD, free extents are considered to start at cylinder 1. If the volume architecture is FBA, free extents are considered to start at block 32.
5. In the normal case, the starting cylinder or block for one minidisk will be greater than the ending cylinder or block for the preceding minidisk on the volume. If the starting cylinder or block for any minidisk is less than or equal to the ending cylinder or block of the preceding minidisk, an exception is flagged. If the two minidisks have the:
 - Same starting and ending location, virtual disk address, and owner, differing only in the system affinity associated with the two entries (with neither system affinity being an asterisk), then the condition is flagged as shared.
 - Same starting and ending location, address, owner, differing only in the system affinity with one of them being an asterisk or if the system affinity values are also the same, then the condition is flagged as an error.
 - Different starting locations, or different ending locations, or different virtual addresses, or different owners, then the condition is flagged as an overlap. If any exceptions are flagged, a DIAGNOSE file will be generated in addition to the requested report file.

GET



VM Systems: All

Purpose

The GET operand of the DIRMAINT command is used to retrieve a copy of a user or profile directory entry. With the entry on your own A-disk, you may introduce new control statements or modify existing ones using an editor. The updated directory entry must be restored to the directory using the REPLACE operand.

Operands

Lock

is used to lock the user ID after the GET processing has completed. This is the default.

NoLock

is used if the request is for informational purposes only. No attempts are made to lock the user or profile entry.

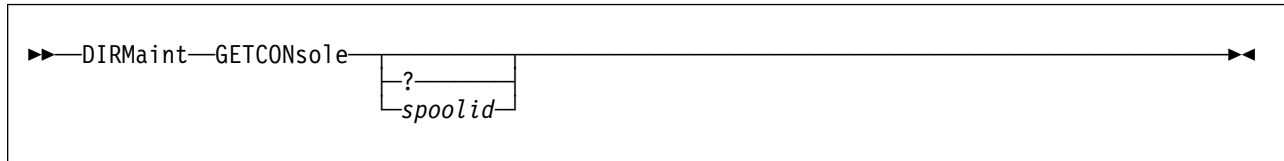
Usage Notes

1. You may use one or more of the prefix keywords (IMMed, REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The entry for this user ID or profile will be sent to your virtual card reader and you will be notified by a message. Use the CMS RECEIVE command to load it onto your minidisk. The file mode allocated defaults to A0.
3. If this request is for information only and you did not use the NOLOCK option, you should immediately issue the UNLOCK operand. On the assumption that the purpose of the GET was to modify some data item, the system has automatically locked the user ID. This locking out is conveyed to you by a message. The user is thus prevented from making any changes while you have a copy of the directory entry for modification.
4. The user ID or profile entry can now be reviewed in your own virtual machine and modified, as appropriate, with the CMS editor.
The revised file can be put into effect by the DIRMAINT REPLACE operand.
5. The lock attempt (which is automatically generated by the GET request) on a user with pending device locks will be rejected. This is done to prevent alterations to the user entry while DASD operations are pending.
6. The default action of a GET request is to issue an implicit LOCK request. This action will succeed if the user issuing the GET is also authorized to

GET

LOCK user IDs. The LOCK transaction will fail, with appropriate error messages, if the user is not authorized for LOCK requests.

GETCONSOLE



VM Systems: All

Purpose

The GETCONSOLE operand of the DIRMAINT command is used to return a copy of the console logfile from the DIRMAINT machine.

Operands

? specifies that the list of available console files is to be displayed to the issuing user.

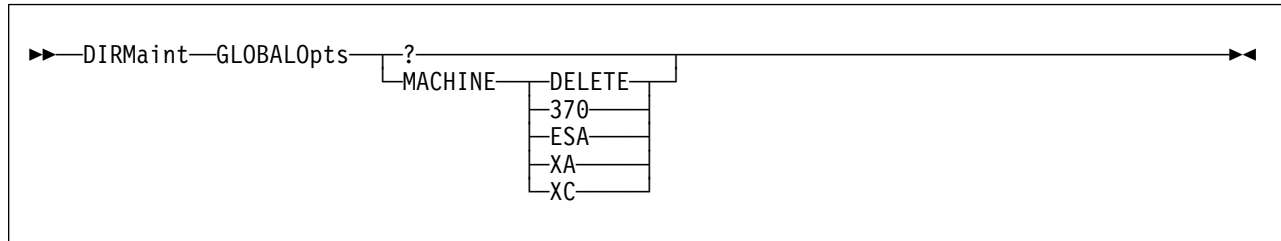
spoolid

specifies the specific spool ID to be returned to the user. If *spoolid* is not specified, the currently active console file is closed and returned.

Note: Specifying the spool ID of the currently active console spool file will result in an error. To obtain a copy of the currently active console spool file, omit the *spoolid* parameter.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

GLOBALOPTS


VM Systems: VM/ESA (ESA Feature)

Purpose

The GLOBALOPTS operand of the DIRMAINT command is used to add, delete, alter, or query the current setting of the GLOBALOPTS control statement.

Operands

? requests that the current settings of the GLOBALOPTS statement be returned in a message.

MACHINE

indicates that one of the following actions should occur on the MACHINE field of the GLOBALOPTS statement:

DELETE

specifies that the MACHINE field be deleted.

370 sets the MACHINE field to 370.

ESA

sets the MACHINE field to ESA.

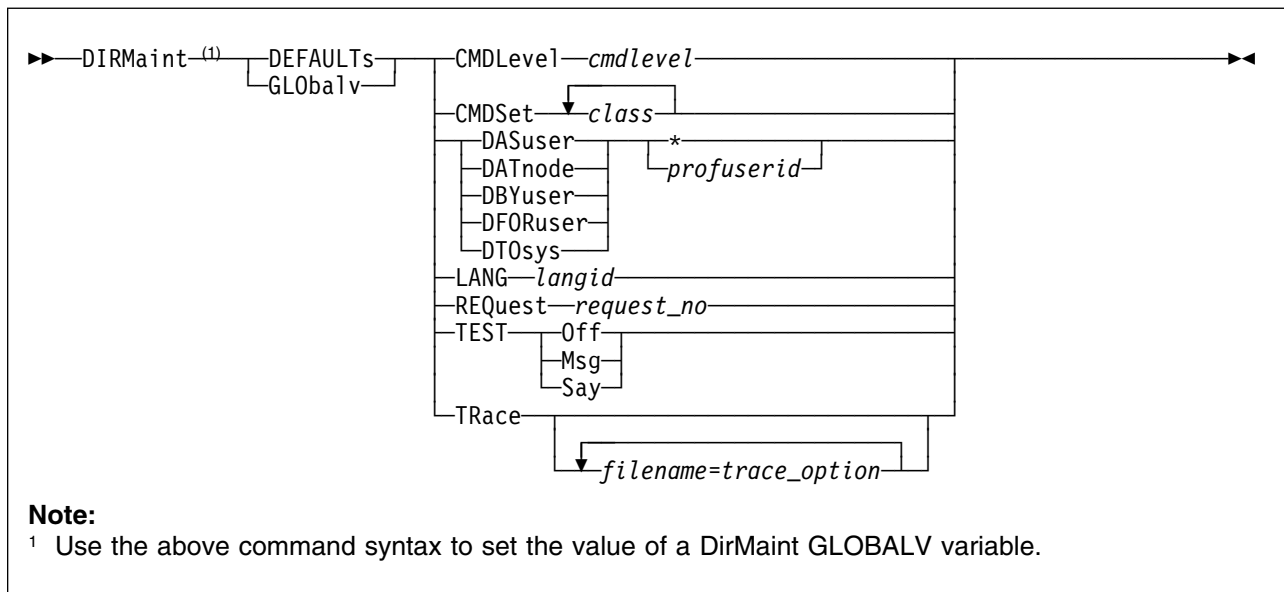
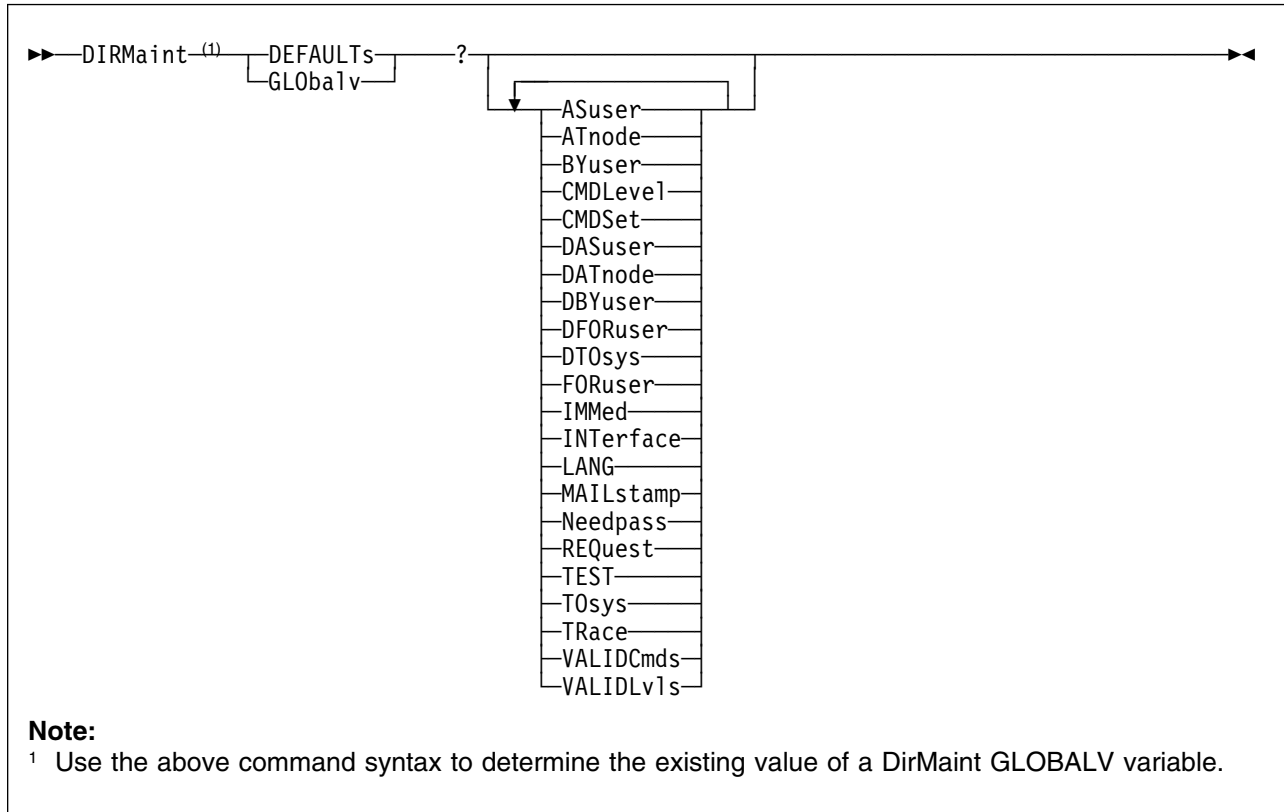
XA sets the MACHINE field to XA.

XC sets the MACHINE field to XC.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. These files will create the \$DIRGRP DIRMPART file if none exists. Also, this file contains the GLOBALDEFS statement that is used to define the GLOBALOPTS and POSIXGROUP statements.
3. For more information on the GLOBALOPTS directory statement, see *z/VM: Planning and Administration*.

GLOBALV, DEFAULTS



VM Systems: All

Purpose

The DEFAULTS or GLOBALV operand of the DIRMAINT command is used to set or query various default values used within DirMaint processing.

Operands

where:

? Causes the current value of one or more variables to be displayed.

ASuser

Is set by the program from the ASUSER prefix keyword on the DIRM command (if specified), or from the DASUSER global variable (if previously set), or to an * by default.

ATnode

Is set by the program from the ATNODE prefix keyword on the DIRM command (if specified), or from the DATNODE global variable (if previously set), or to an * by default.

BYuser

Is set by the program from the BYUSER prefix keyword on the DIRM command (if specified), or from the DBYUSER global variable (if previously set), or to an asterisk (*) by default.

CMDLevel *cmdlevel*

May be set by the user to indicate the compatibility level desired for use with application programs. Valid *cmdlevel* values are 150A or 140A.

CMDSet *class(es)*

May be set by the user to indicate the command set the user believes is authorized for his or her use. This is expressed as a single string of 1 or more characters. The valid *class* values are tailorable by the installation. The IBM supplied default command classes are:

- G for General Users
- A for System Administrators
- D for DASD Management Staff
- H for the Help Desk Team
- M for password monitoring commands
- O for the System Operator
- P for automated programs such as DFSMS/VM®
- S for the System Programmer
- Z for internal use between the DIRMAINT and DATAMOVE virtual machines

Up to 36 characters may be specified, in any order, with no spaces in between them.

The DirMaint Release 5 classes equate to the DirMaint Release 4 authorities as follows:

| DirMaint Release 5 classes | DirMaint Release 4 authority |
|---|---|
| G S | OWNER |
| G O | SYS_OPER |

| | |
|-----------|----------------|
| G A D H P | DIRM_STAFF |
| G H | DIRM_SUB_STAFF |
| G M | PWMON |

DASuser

Identifies the default value for the ASUSER prefix keyword on the DIRMAINT command. Valid values are an asterisk or a valid file name which may be used as a user ID.

DATnode

Identifies the default value for the ATNODE prefix keyword on the DIRM command. Valid values are an asterisk or a valid file name which may be used as a network node ID.

DBYuser

Identifies the default value for the BYUSER prefix keyword on the DIRM command. Valid values are an asterisk or a valid file name which may be used as a user ID.

DFORuser

Identifies the default value for the FORUSER prefix keyword on the DIRM command. Valid values are an asterisk or a valid file name which may be used as a user ID.

DTOsys

Identifies the default value for the TOSYS prefix keyword on the DIRM command. Valid values are an asterisk or a valid file name which may be used as a network node ID.

IMMed

Is set by the program based on the presence or absence of the IMMED keyword prefix on the DIRM command.

LANG *langid*

A valid *langid* consists of 5 characters or less, that are valid in a CMS file name, and is explicitly defined on one or more *langid* USER MSGS entries in the CONFIG* DATADVH file(s).

Also a *langid* of *1SAPI* may be used to specify the *application programming interface* language, where all replies are:

```
REQUEST= nnnn RTN=DVHXXX MSGNO=1234 FMT=01
SUBS= VAR1 VAR2
```

MAILstamp

Displays the last time the mail file was sent to you.

Needpass

Is set by the NEEDPASS command to indicate whether password prompting is done for most DirMaint commands (YES) or not (NO).

REQuest *request_no*

Specifies an identifying number to be associated with a transaction sent to the DIRMAINT server. Used by programs using LANG 1SAPI to associate the transactions with the returned messages.

TEST

Determines whether or not a DIRM command results in a transaction to the DIRMAINT service machine (TEST OFF), or is displayed on the invoker's

GLOBALV

console as a CP message (TEST MSG), or is displayed on the invoker's console using a CMS REXX SAY statement (TEST SAY). Valid values are OFF, MSG, or SAY.

TOSys

Is set by the program from the TOSYS prefix keyword on the DIRM command (if specified), or from the DTOSYS global variable (if previously set), or to an * by default.

TRace

Is used for problem diagnosis to activate REXX tracing for the listed routines, and specify the degree of detail desired in the trace. Each entry must be in the form of a file name, followed by an equal sign, followed by a REXX trace option (All, Commands, Error, Failure, Intermediates, Labels, Normal, Off, Results, Scan); with no intervening blanks.

VALIDCmds

Is set by the program to identify the valid values for CMDLEVELS. The valid values are 150A and 140A.

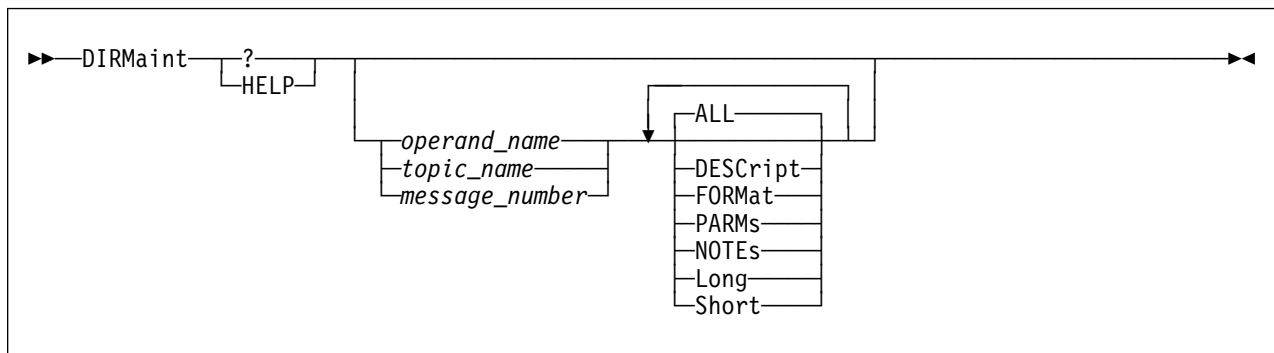
VALIDLvls

Is set by the program to identify the valid values for the interface level. The current value is 199501.

Usage Notes

1. DirMaint prefix operands do apply to this command. They may be accepted but will be ignored.
2. The DEFAULTS and GLOBALV commands are completely handled within the virtual machine of the command issuer. No transaction is sent to the DIRMAINT service machine, therefore no password is required or requested for authentication, even if you have set NEEDPASS to YES.
3. For more information on altering the NEEDPASS setting, see "NEEDPASS" on page 156.
4. The DEFAULTS and GLOBALV commands are not valid within a BATCH command file.
5. All global variables are stored in the DVH15 variable pool.

HELP



VM Systems: All

Purpose

The HELP operand of the DIRMAINT command is used to obtain online information about a DirMaint operand or message.

Operands

operand_name

Is the name of any DirMaint operand on which information is desired. The available operands can be found by omitting the operand name and selecting one of the choices shown on a menu.

topic_name

Is the name of a DirMaint topic other than a DIRMAINT command on which help is desired. The available topics can be found by omitting the topic name and selecting one of the choices shown on the menu.

message_number

Is the identification of a DirMaint message.

ALL

Displays all information available about the chosen operand, topic, or message.

DESCRiption

Displays general description information only. See Usage Note 3 on page 122.

FORMat

Displays format or syntax information only. See Usage Note 3 on page 122.

PARMs

Displays parameter information only. See Usage Note 3 on page 122.

NOTEs

Displays Usage Notes only. See Usage Note 3 on page 122.

Long

Displays all HELP information available for the specified command, message, or topic. This is the same as specifying ALL.

Short

Displays an abbreviated version of the HELP file if it exists. Otherwise, the entire HELP files is displayed.

Note: DirMaint Release 5.0 does not currently provide abbreviated HELP files.

Usage Notes

1. DirMaint prefix operands do apply to this command. They may be accepted but will be ignored.
2. All DirMaint messages are prefaced with a message identification in the form of DVH*rrrrnnns*. When specifying the message number to obtain help, only the DVH prefix and *nnnn* message number should be used. For example, use DIRM ? DVH1232 for help on message DVHCMD1232E or DVHPAR1232E.

Note: When using this form of the command, the prefix (DVH) and any suffix (E in this case) is optional.

Outside of DirMaint you may also use CMS HELP xDVH DVH1234 for help on DVHCMD1234E or DVHPAR1234E, where x is A for AMENG, U for UCENG, or a unique letter to identify other languages.

3. When HELP is used for other than an operand name, the result is the same as using ALL.
4. The HELP command is completely handled within the virtual machine of the command issuer. No transaction is sent to the DIRMAINT service machine, therefore no password is required or requested for authentication, even if you have used DIRM NEEDPASS to set NEEDPASS to YES.
5. The HELP command is not valid within a BATCH command file.
 - li.Presuming that the DirMaint HELP files are installed on the same disk as the CMS and CP system HELP files, or that they are on an accessed minidisk or SFS directory, then the same result can be obtained by using the CMS system HELP command. Using the CMS HELP command requires specification of the language and component identification. Thus, DIRM ? MDISK becomes HELP ADVH MDISK for help in mixed case American English, or HELP UDVH MDISK for help in upper case English. The last three characters of the component code will be DVH for all DirMaint help files; the first character will usually be the initial letter of the name of the language. Check with your local support group to determine which languages are available, and the prefix letter for each.

Examples

1. Invoking all HELP information for the DIRMAINT APPCPASS operand.

```
DIRMAINT ? APPCPASS
```
2. Invoking only the syntax diagram for the DIRMAINT LINK operand:

```
DIRM HELP LINK FORMAT
```
3. Invoking all HELP information available for the DIRMAINT USEREXIT topic:

```
DIRM HELP USEREXIT
```
4. Invoking the HELP information for DIRMAINT message number DVH2119:

```
DIRM HELP DVH2119
```

HISTORY

▶▶—DIRMaint—HISTory—▶▶

VM Systems: All

Purpose

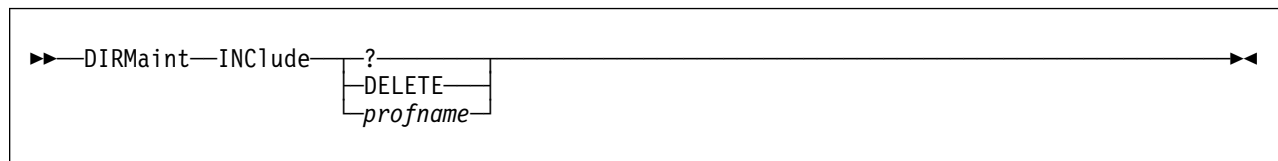
The HISTORY operand of the DIRMAINT command is used to request a copy of the history file.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The history file is a time-stamped sequential listing of all incoming operands.
3. The LOGMSG operand may be used to insert supplementary messages into the history file. For more information, see “LOGMSG” on page 135.
4. The disposition of older versions of the history file is determined in the DVHDAILY EXEC procedure.
5. The history file is identified on the MESSAGE_LOGGING_FILETYPE= configuration file entry.

INCLUDE

INCLUDE



VM Systems: All

Purpose

The INCLUDE operand of the DIRMAINT command allows the caller to query, alter, or delete the current INCLUDE directory statement.

Operands

? allows you to query the current INCLUDE statement.

DELETE

request to delete the current INCLUDE statement.

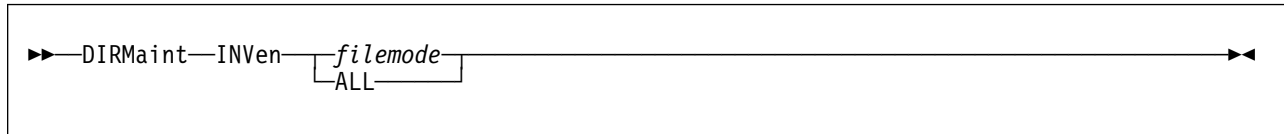
profname

the name of the profile to be placed on the new INCLUDE statement.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the INCLUDE directory statement, see *z/VM: Planning and Administration*.
3. Because DELETE is a keyword, it cannot be used as a profile name. To use DELETE as a profile name, use the DIRM GET and DIRM REPLACE operands.

INVEN



VM Systems: All

Purpose

The INVEN operand of the DIRMAINT command is used to obtain an inventory of one or all of the DIRMAINT virtual machine's minidisks.

Operands

filemode

specifies which of the minidisks you want to inventory.

ALL

Requests that all minidisks accessed R/W be inventoried.

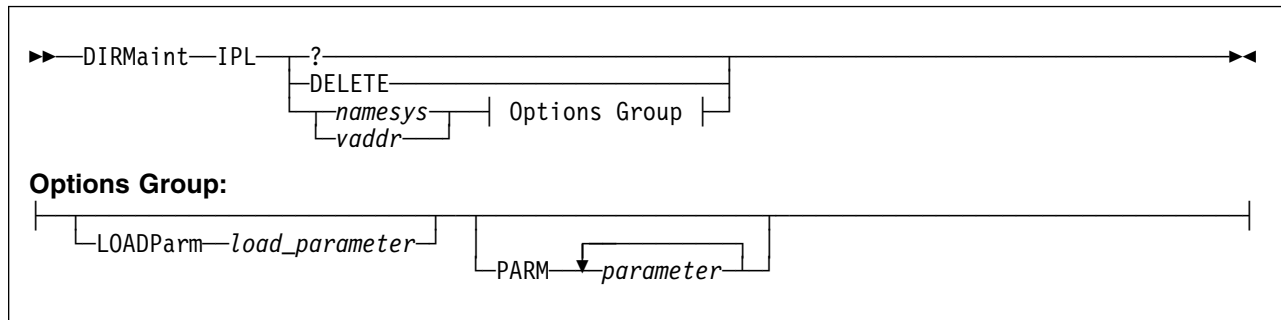
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The specified disk will be inventoried by executing the

```
CMS LISTFILE * * filemode ( DATE
```

 command.
3. The resultant file DIRMAINT DISKx will be sent to your virtual card reader. Use the CMS RECEIVE command to read the file.
4. If the disk is a CMS minidisk, a QUERY DISK is issued against the disk and appended to the file. If the disk is a SFS directory, a QUERY ACCESSED is issued against the file mode and appended to the file.
5. If the ALL operand is used, you will receive a separate file for each disk.

IPL



VM Systems: All

Purpose

The IPL operand of the DIRMAINT command is used to add, delete, query, or change your directory entry IPL control statement. This statement specifies that CP automatically load your virtual machine when you log on to the system. The PARM option may be included for passing parameters to the IPL'd system.

Either *namesys* or *vaddr* will add an IPL control statement to your directory entry or will replace a statement that currently exists.

Operands

? requests that the current setting of your IPL statement be queried.

DELETE

requests that the current IPL statement is to be deleted.

namesys

is the name of a saved system to be automatically loaded each time you log on.

vaddr

is the virtual device address of the device containing the system to be loaded.

LOADParm *load_parameter*

specifies a 1 to 8 character load parameter that is used by the IPL'd system.

PARM *parameters*

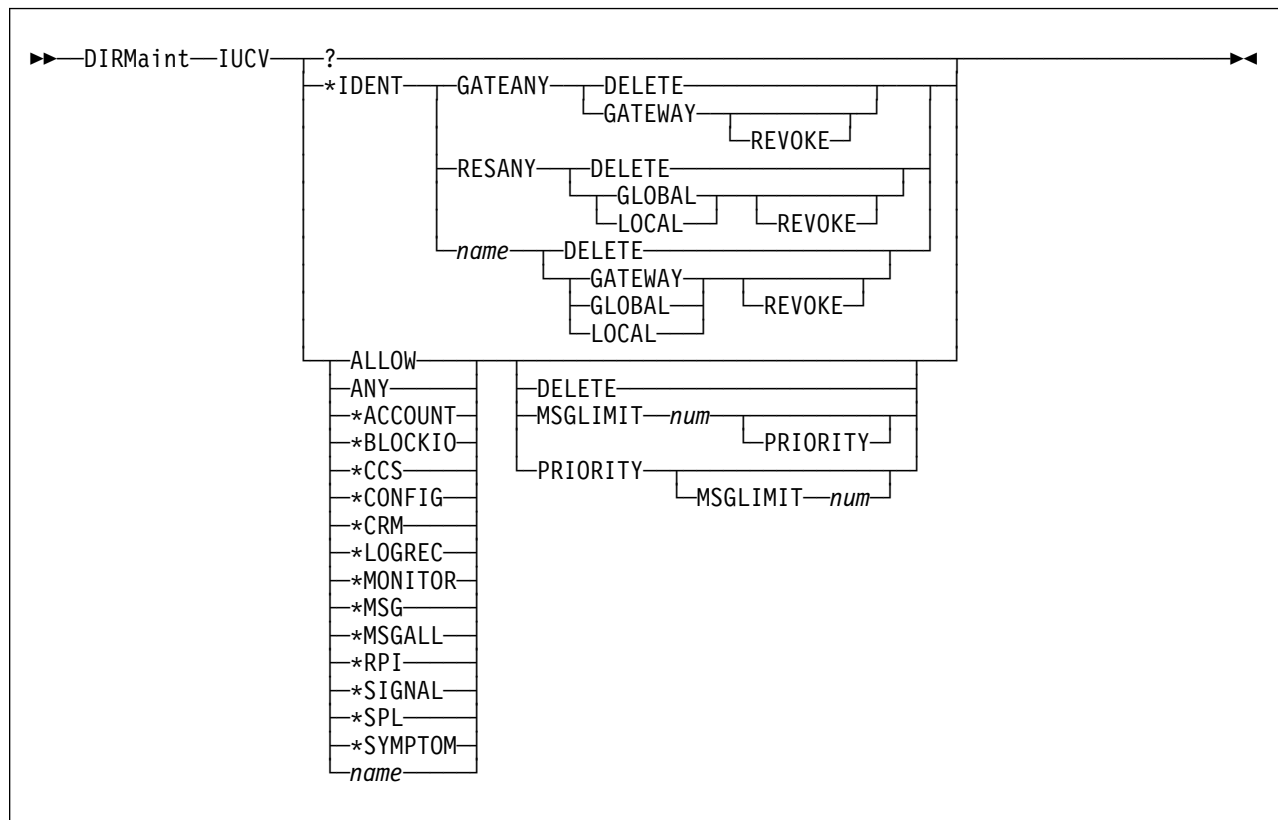
specifies the parameters to be passed to the IPL'd operating system, separated by blanks. Refer to Usage Notes 4, 5, and 8.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the IPL directory statement, see *z/VM: Planning and Administration*.

3. The directory IPL control statement can be overridden at logon time by specifying LOGON *userid* NOIPL.
4. If multiple blanks are inserted between parameters, they are compressed to one blank.
5. The entire PARM string is placed into the directory provided it fits in 72 columns. Characters after column 72 are truncated.
6. Since the parameters specified on the PARM parameter are only meaningful to the system to be loaded, DirMaint does not validate them. The data is passed to the operating system that is receiving control as a result of the IPL process.
7. Because DELETE is a DirMaint keyword, it cannot be used as the name of a saved system on the IPL command. To use DELETE as a saved system name, use GET and REPLACE processing.
8. Because DIAGNOSE X'84' processing handles only the first 48 bytes of information, the PARM string is left justified and truncated to 48 bytes before being passed to the DIAGNOSE X'84' routine.
9. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

IUCV



VM Systems: All

Purpose

The IUCV operand is used to add, delete, change, or query your directory IUCV control statements.

Operands

? requests that the current settings of your IUCV statements be queried.

*IDENT

allows this virtual machine to connect to the Identify System Service to identify a resource.

GATEANY

allows the virtual machine to identify any gateway LU name.

RESANY

allows the virtual machine to identify any resource name.

name

a 1 to 8 character name representing the gateway name, resource ID, gateway LU or user ID associated with the IUCV statement.

DELETE

requests that the specified IUCV statement be deleted.

GATEWAY

authorizes the virtual machine to identify the gateway LU.

LOCAL

authorizes the virtual machine to identify the resource as a local resource.

GLOBAL

authorizes the virtual machine to identify the resource as a global resource.

REVOKE

allows the virtual machine to revoke the specified resource or gateway LU name without owning it.

ALLOW

specifies that any other virtual machine can establish a communication path with this virtual machine.

ANY

is a general authorization indicating that a communications path can be established with any other virtual machine, resource, or gateway on this VM/ESA system.

***ACCOUNT**

Accounting system service.

***BLOCKIO**

DASD block I/O system service.

***CCS**

SNA Console Communications Service system service.

***CONFIG**

Configuration system service.

***CRM**

Collection resource management system service.

***LOGREC**

Error recording system service.

***MONITOR**

*MONITOR system service.

***MSG**

Message system service.

***MSGALL**

Message all system service.

***RPI**

Access verification system service.

***SIGNAL**

Signal system service.

***SPL**

Spool system service.

***SYMPTOM**

Symptom system service.

IUCV

MSGLIMIT *num*

defines the maximum number of outstanding messages allowed on any path authorized by this entry.

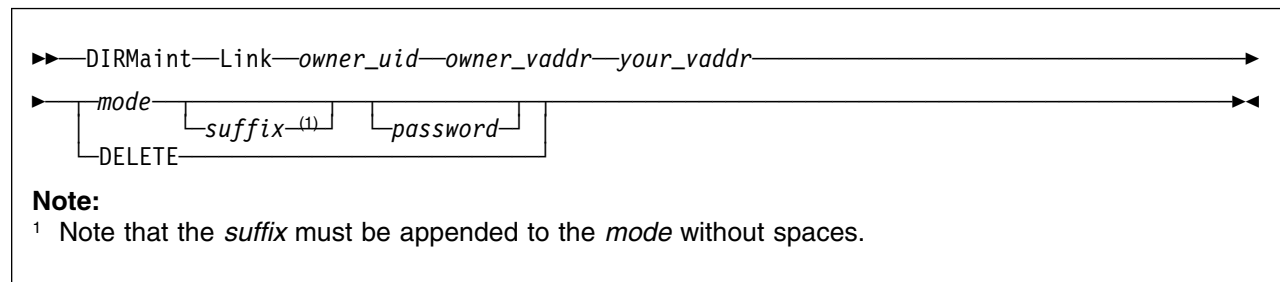
PRIORITY

indicates that a communication path with the specified virtual machine can handle priority IUCV interrupts.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Not all options of this command are appropriate for VM/ESA (ESA Feature). For more information on the IUCV directory statement, see *z/VM: Planning and Administration*.
3. As the IUCV statement can be used to control communication between virtual machines, careful consideration must be given to the security implications of using this statement.

LINK



VM Systems: All

Purpose

The LINK operand allows you to request that a LINK control statement be inserted in your directory entry and put into effect automatically each time you log on. This makes it possible for your virtual machine to automatically establish a link to another virtual machine's minidisk without you needing to issue the CP LINK command during your terminal session.

Operands

owner_uid

specifies the user ID of the minidisk owner to which the requestor wishes to link.

owner_vaddr

specifies the virtual address of the minidisk as addressed by the owner.

your_vaddr

specifies the virtual device address you will use to access the minidisk.

mode

specifies the type of access you have to the minidisk when you log on. The owner must have assigned a password corresponding to the mode you are requesting or must have granted specific permission to link to the device using the AUTHLINK command. For example, if the owner has established a read password or authorized a read link, then you can request a mode of R or RR. Valid modes are:

Mode Definition

- R** Read-only (R/O) access is desired. Access is not allowed if the owner or any other user has a link to the minidisk in write or any exclusive status.
- RR** Read-only (R/O) access is desired, even if the owner or another user has a link to the minidisk in write status. Access is denied if any user has the minidisk linked in exclusive status.
- W** Write access is desired. The minidisk is not accessible if the owner or any other user has a link to the minidisk in read or write status.
- WR** Write access is desired. Only R/O access is allowed if the owner or any other user has a link to the minidisk in read or write status. Access is denied if any exclusive links exist.

LINK

- M** Multiple access is desired. A write link is allowed to the minidisk unless another user already has write, stable or exclusive access to it, in which case, the minidisk is not accessible to you.
- MR** Write or any exclusive access is allowed to the minidisk unless another user already has write access to it, in which case R/O access is allowed to the minidisk. Access is also denied if any exclusive links exist.
- MW** Write access is allowed to the disk unconditionally, except for existing stable or exclusive links. Access is denied if any stable or exclusive links exist.

suffix

specifies an optional mode suffix letter to be directly appended (without spaces) to the access mode. Valid suffix letters are:

Suffix Definition

- E** exclusive access is desired.
- S** stable access is desired.

password

defines the virtual machine owner's access password for the target minidisk. It must correspond to the mode of access that you are requesting. You may enter it as the last operand of the command as long as you take appropriate security precautions. It is recommended that you omit this operand and enter the password with display inhibited after the system requests it.

DELETE

specifies that you are voluntarily dropping the link.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode, PROMPT) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Use of the PROMPT prefix operand will result in command line prompting for a missing password.
3. For more information on the LINK directory statement, see *z/VM: Planning and Administration*.
4. The owner becomes aware of your link the next time they issue the REVIEW operand and may decide to drop the link.
5. If the owner of the target minidisk has disabled their virtual machine for the DIRM LINK operand (by means of the DIRM USEROPTN LINKS DISABLE command), then you will not be able to establish a link to the minidisk even if you know the password; unless you have been specifically authorized to link to the device (by a DIRM AUTHLINK command) or you are authorized to act for ALL users. For more information, see "AUTHLINK" on page 39.
6. The access password must be supplied but need not be correct if it is ALL or if the transaction originator is authorized to act for ALL users or you

have been previously authorized to establish the link. For more information, see "AUTHLINK" on page 39.

7. The maximum number of indirect links allowed can be set by the LINK_MAX_INDIRECT= setting in the CONFIG* DATADVH. If not specified, The default for VM/ESA (ESA Feature) is 50.

An indirect link is a link to a LINK statement in another directory entry which in turn links to a minidisk. Consider the following example:

If the following statements are in the indicated directories:

USERA:

```
MDISK 191 3380 1 10 MJHVOL ALL ALL ALL
```

USERB:

```
LINK USERA 191 192 RR
```

USERC:

```
LINK USERB 192 193 RR
```

USERA owns the actual minidisk statement as their 191 disk. USERB has a direct link to it as their 192 disk. USERC has an indirect link to USERA's disk as their 193 disk.

If the number of indirect links exceeds the current maximum allowed indirect link count, the link attempt is rejected with a message.

If the command handler is unable to follow the indirect chain to the root minidisk statement, the LINK attempt will be rejected with a message.

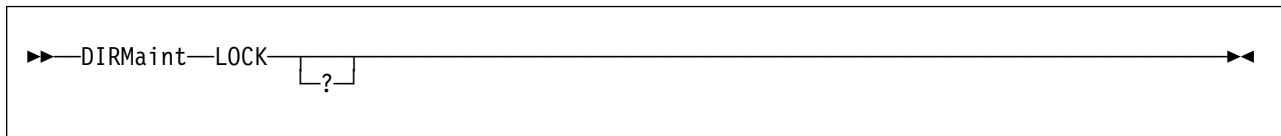
8. If the DIRM LINK threshold function is enabled, the LNKLMT= statement (located in the CONFIG* DATADVH file) attempts to establish a threshold for failed link attempts. A file named DVHLINK ATTEMPT contains an entry for each user who has issued an invalid or incorrect DIRM LINK operand. Each entry includes the number of unsuccessful link attempts and the date the last unsuccessful attempt was made.

When the counter of invalid attempts for a user reaches the specified threshold, a message containing this information is sent to the user and the link attempt is rejected.

9. The DVHLINK ATTEMPT file can be administered using the DIRM ELINK operand.

LOCK

LOCK



VM Systems: All

Purpose

The LOCK operand of the DIRMAINT command is used to temporarily prevent a user from making any changes to their directory entry. It also inhibits retrieval or modification of that entry by any other user. When automatically invoked by the GET operand, the LOCK operand also locks profile entries.

Operands

? requests that the current status of the user ID be returned.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The LOCK operand can be used whenever it is appropriate to inhibit you from making any changes directly to your own directory entry.
3. The LOCK operand will be invoked automatically whenever the GET operand is used unless the NOLOCK option is used with GET. Normally this is the only case that you would use it to lock a profile entry.
4. A lock request is refused with messages if a user lock already exists for the user ID, or if any minidisk device locks are currently enabled for the user ID.

LOGMSG

►►—DIRMaint—LOGmsg—*msgtext*—————◄◄

VM Systems: All

Purpose

The LOGMSG operand of the DIRMAINT command is used to place a note into the history file that is maintained by the DIRMAINT virtual machine. A message that contains text from the note is issued to the console.

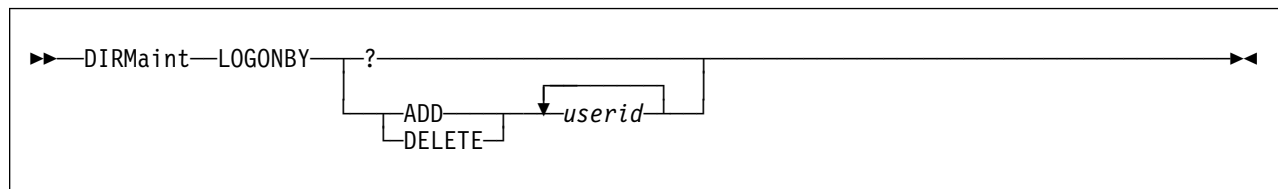
Operands

msgtext

is the message text that is to be recorded in the DirMaint history file.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. All transactions and other significant events are recorded by DirMaint. This operand can be used to place a note into this file, as a historical record. The history file is a daily log. Disposition of the current log at the end of a day is an installation option. The current history log can be retrieved at any time. For more information, see“HISTORY” on page 123 .

LOGONBY


VM Systems: VM/ESA (ESA Feature) 1.2.2 and later

Purpose

The LOGONBY operand of the DIRMAINT command is used to add or delete users from the current LOGONBY directory statement. To determine the current users on the list, a query option is supported. The LOGONBY directory statement lists users that can use their own passwords to logon to different user IDs.

Operands

? indicates that a query request is being made.

ADD

DELETE

indicates that the listed user IDs are to be added to, or deleted from the list currently in the directory.

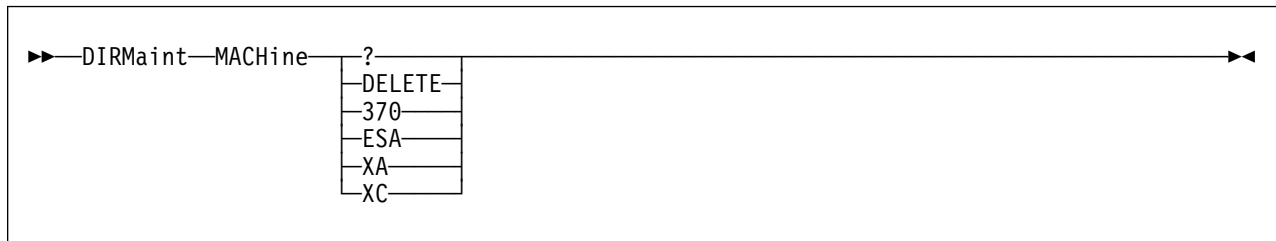
userid

a 1 to 8 character user ID that is to be added to or deleted from the LOGONBY directory statement.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the LOGONBY directory statement, see *z/VM: Planning and Administration*.
3. Attempts to delete a non-existent user ID will result in an error message and the command will be rejected.
4. Attempts to add a user ID that already exists will result in an error message and the command will be rejected.
5. If a LOGONBY directory entry does not exist, one will be created.
6. A maximum of eight users can exist on the LOGONBY directory statement in the user directory.

MACHINE



VM Systems: VM/ESA (ESA Feature)

Purpose

The MACHINE operand of the DIRMAINT command allows the general user to create or change a MACHINE statement in a directory entry. If a MACHINE statement does not exist in your directory entry, a new statement is created. If a MACHINE statement does exist, the machine type is changed as specified in the MACHINE operand.

Operands

? requests a query be done of the current MACHINE statement.

DELETE

requests that the MACHINE statement be deleted from the directory.

370

defines a 370-mode virtual machine.

ESA

defines an ESA-mode virtual machine.

XA

defines an XA-mode virtual machine.

XC

defines an XC-mode virtual machine.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For a complete description of the MACHINE directory statement, see *z/VM: Planning and Administration*.
3. For more information about the various modes of operation, see the CP SET command in the *z/VM: CP Command and Utility Reference*.
4. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

MAIL

▶▶—DIRMaint—MAIL—————▶▶

VM Systems: All

Purpose

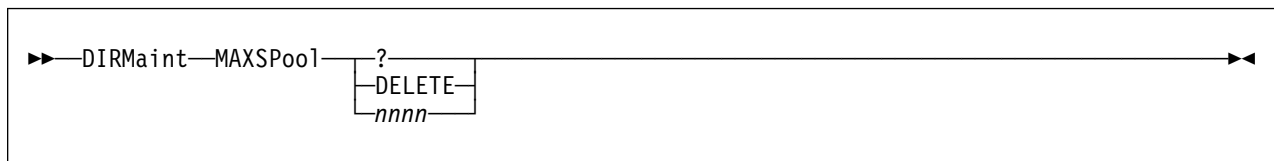
The MAIL operand of the DIRMAINT command causes a specific file (DIRMAINT NEWMAIL A) to be forwarded to DirMaint and stored on DirMaint's user interface disk.

This operand is intended for use by system administrators who wish to forward mail to all DirMaint users. The mail file may contain DirMaint system information about new or changed function or installation policy. Mail recipients are advised to load mail from their virtual reader using RECEIVE to review the information.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. This operand can be used to send a MAIL file to users of DirMaint. Actually, the MAIL file will be stored in the DIRMAINT virtual machine, and will be disseminated to users when they next invoke a DIRMAINT operand.
3. This is a mechanism whereby users can be notified of changes, new function, new policies, and so on. Once a MAIL file has been set up, users who subsequently invoke a DIRMAINT operand will automatically receive a copy of the MAIL file in their virtual card readers. Any given user with a R/W A-minidisk or directory will be sent only one copy of the MAIL file.
4. Prior to invoking this operand, you, as the system administrator, must have used an editor, in your own virtual machine, to create the file DIRMAINT NEWMAIL on your A-minidisk.

MAXSPOOL



VM Systems: VM/ESA (ESA Feature)

Purpose

The MAXSPOOL operand of the DIRMAINT command is used to add, delete, change, or query a SPOOLFILE directory control statement.

Operands

? specifies a request to query the current SPOOLFILE statement.

DELETE

specifies a request to delete the SPOOLFILE statement from the user's directory entry.

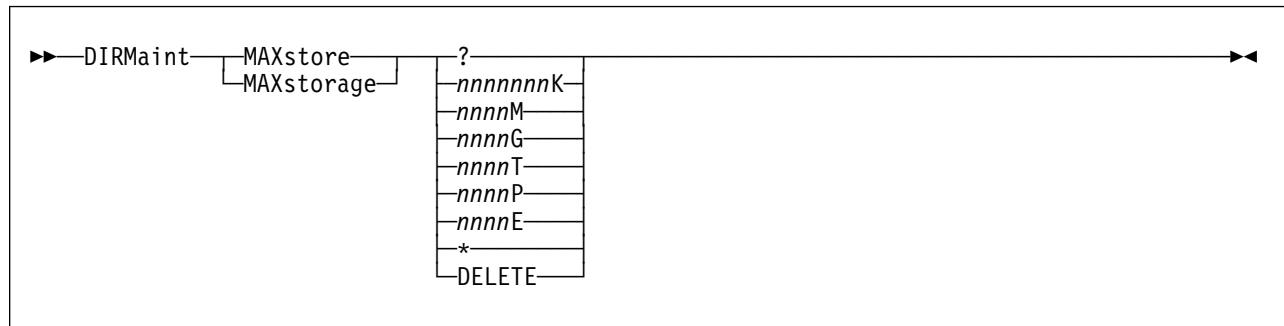
nnnn

specifies the number of spool files that the subject user is allowed to have. This number can range from 1 to 9999.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For a complete description of the SPOOLFILE directory statement, see *z/VM: Planning and Administration*.
3. If no SPOOLFILE MAXSPOOL statements are specified, the CP default of 9999 spool files is in effect.
4. If a SPOOLFILE statement does not exist in the directory definition and DELETE is not specified, a new statement is created. If such a statement exists and DELETE is specified, the statement is removed. If a SPOOLFILE statement exists and DELETE is not specified, the supplied value replaces whatever was previously on the record after appropriate validation is complete.

MAXSTORE, MAXSTORAGE



VM Systems: All

Purpose

The MAXSTOR operand of the DIRMAINT command is used to change the maximum virtual storage size that a user ID can define.

Operands

? specifies that this transaction is a query to determine the current maximum virtual storage size for this user.

nnnnnnnK
specifies the maximum value allowed in kilobytes.

nnnnM
specifies the maximum value allowed in megabytes.

nnnnG
specifies the maximum value allowed in gigabytes.

nnnnT
specifies the maximum value allowed in terabytes.

nnnnP
specifies the maximum value allowed in petabytes.

nnnnE
specifies the maximum value allowed in exabytes.

***** when used within a:

- USER directory entry, specifies that the value specified within the included PROFILE entry will be used if present, or the CP default value of 1M will be used if there is no MAXSTORAGE value within the included PROFILE; and
- PROFILE directory entry, specifies that the CP default value of 1M will be used unless the including USER entry specifies an overriding MAXSTORAGE value.

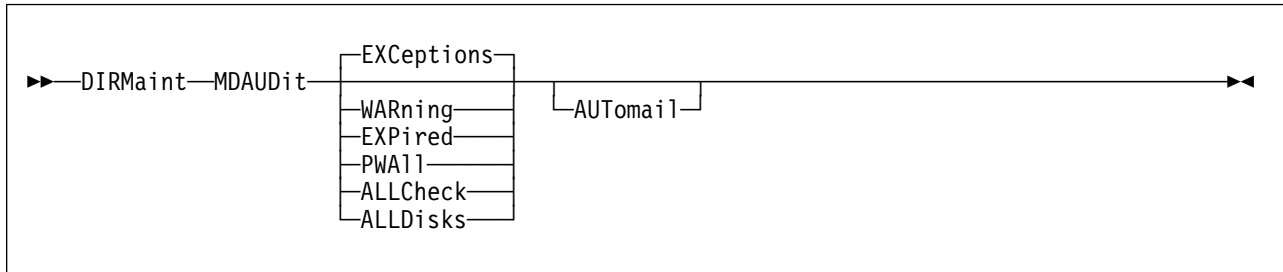
DELETE

specifies that the MAXSTORAGE statement will be deleted, resulting in use of the:

- Value specified within an included PROFILE if deleted from a USER directory entry.
- CP default value of 1M when deleted from a PROFILE entry, unless the including USER entry specifies an overriding MAXSTORAGE value.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the USER directory statement, see *z/VM: Planning and Administration*.
3. The value is placed into the directory as provided once it has been validated for correct format. Any required rounding is done by CP.
4. If the directory definition of the user ID being modified does not currently have default or maximum virtual storage sizes defined, a default logon storage size of 256KB is assigned to the user ID. This value becomes the minimum that can be specified for the maximum virtual storage size.
5. The values supplied must be whole numbers. Values such as 4096.8K or 6.5M are not supported.
6. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.
7. The suffix K, M, G, T, P or E is required.
8. A query request will report the default values for the default storage size and maximum storage size if they are not specified on the user statement.

MDAUDIT


VM Systems: All

Purpose

If an installation is using the MDISK password monitoring function, the MDAUDIT operand of the DIRMAINT command checks for compliance with established policy.

Operands

EXCeptions

all disks showing an exception (password of ALL, expired, or in the warning period) are presented. This is the default.

WARning

all minidisks that are considered to be expired or in the warning period are presented.

EXPIred

only disks with expired passwords are presented in the report.

PWAll

produces a report that is sent to the invoker as a spool file listing all instances of the password ALL. Two circumstances are recognized:

1. ALL used as a READ password. This is a concern only from a security standpoint, and is usually done deliberately to effectively make the contents of a disk "public."
2. ALL used for a WRITE or MULTIPLE access password. If this is done by a general user, it is probably a mistake. A serious data integrity problem can develop, since any other user can write and erase files on such a disk.

ALLCheck

all disks showing an exception (password of ALL, expired, or in the warning period) are presented. This option is synonymous with option EXCEPTIONS.

ALLDisks

all disks that have minidisk passwords are presented. Even disks without any exceptions are reported.

AUTomail

sends electronic mail directly to the users who own minidisks for which exceptions are detected. This “form letter” is described under Usage Notes 5, 6, 7, and 8.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. MDISK Password Monitoring is a built-in function that is inactive by default. Installations that want to ensure that MDISK passwords are changed periodically may define the appropriate frequency in the DirMaint configuration file. An example follows:

```
MDPW_INTERVAL= ww ll
```

ww represents the time (in days) until the user is considered to be in the WARNING period. *ll* represents the time (in days) until the user is considered to have expired passwords. Values for these intervals should be in the range:

```
0 <= ww <= ll <= 366
```

If this entry is not found, if both values are not provided or the values are in an invalid format, the MDAUDIT command is rejected with a message.

DirMaint time stamps the password changes on a minidisk basis. Three time stamps are maintained; one each for READ, WRITE, and MULTI.

3. Once minidisk password time stamping is in effect, you may begin using the MDAUDIT function to detect instances of non-compliance with your defined policy. It will only report exceptions as detected. Installations will develop their own appropriate means of dealing with failure to adhere to password management policy.
4. The normal output of the MDAUDIT function is a spool file returned to the invoker, which should be loaded from the virtual reader by the CMS RECEIVE command. This file contains one record for each minidisk for which some exception has been noted. The passed parameters affect the content of the report. This file is intended to be visually readable, as well as machine readable. Since it is expected that most installations will create some sort of automatic procedure to process this file, column headings are not included. To aide in the development of an application, all fields are filled in with one or more “-” characters if they are not meaningful to the exception report. The returned file has the following fields in the specified order:

Field Name Meaning and value

User ID The user ID that owns the mdisk device.

SysAffin The System Affinity of the device.

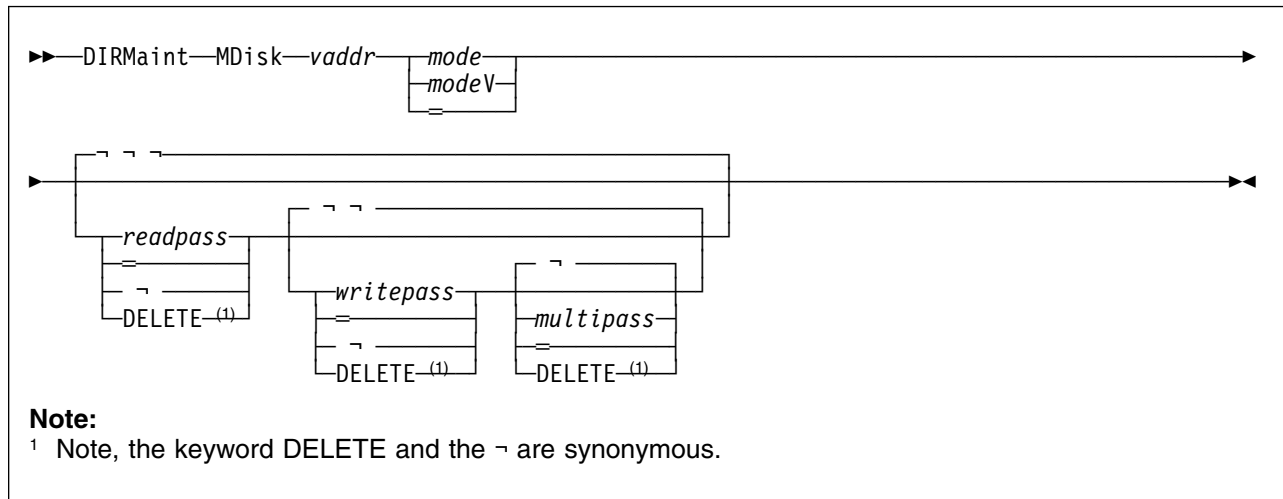
Ccuu The address of the device.

| | |
|--------------|--|
| Rtype | The type of READ exception associated with this device. Valid exceptions for READ, WRITE, and MULTI include: Exception Meaning NONE No exception noted for this password. WARN This password is in the warning period. *EXP This password is expired. *ALL This password was found to be set to ALL. |
| Rcd | The READ change date. With the format yyyy/mm/dd |
| Rdsc | The number of days since the READ password was changed. |
| Rdod | The number of days a READ password change is overdue, or the string "---" if the password change is not overdue. |
| Rdtcd | The number of days until a READ password change is due, or the string "---" if the password is currently overdue. |
| Wtype | The type of WRITE exception associated with this device. See READ above for detailed meanings of exception types. |
| Wcd | The WRITE change date. With the format yyyy/mm/dd |
| Wdsc | The number of days since the WRITE password was changed. |
| Wdod | The number of days a WRITE password change is overdue, or the string "---" if the password change is not overdue. |
| Wdtcd | The number of days until a WRITE password change is due, or the string "---" if the password is currently overdue. |
| Mtype | The type of MULTI exception associated with this device. See READ above for detailed meanings of exception types. |
| Mcd | The MULTI change date. With the format yyyy/mm/dd |
| Mdsc | The number of days since the MULTI password was changed. |
| Mdod | The number of days a MULTI password change is overdue, or the string "---" if the password change is not overdue. |
| Mdtcd | The number of days until a MULTI password change is due, or the string "---" if the password is currently overdue. |
| Rall | This field will contain an "X" if the READ password is ALL, it will contain a "-" if it is not. |
| Wall | This field will contain an "X" if the WRITE password is ALL, it will contain a "-" if it is not. |

5. When the AUTOMAIL parameter is specified, a form letter is sent to users with detected exceptions as listed in the "Type of Exception" column.
6. DirMaint Release 5.0 is shipped with a default form letter to be sent to offenders when AUTOMAIL is used. The first two lines of the output are automatically generated by the MDAUDIT processor and contain the user

ID of the individual receiving the letter and the date on which the MDAUDIT program was run. The form letter is drawn from the AUTOMAIL DATADVH file. This form letter should be customized to reflect the correct administrator name and phone extension. When altering this file do not alter the prefix tags that start each line of the report. The MDAUDIT processor uses these to assign position within the final output file.

7. Leap years are not accounted for during date subtraction. February is assumed to always have 28 days, regardless of year.
8. The report sent to the invoker is terminated with a line containing:
END-REPORT
9. If a value exceeds 999, it is set to 999 in the report.
10. AUTOMAIL sends the generated form letter to
*userid AT **

MDISK


VM Systems: All

Purpose

The MDISK operand of the DIRMAINT command allows you to change the access mode and the access passwords for one of your minidisks.

Operands

vaddr

is the virtual device address of the minidisk being changed.

mode

modeV

specifies the user's mode of access at logon. If the minidisk owner allows other users access to the minidisk by establishing access passwords, then the access conditions may be modified by such links. An optional "V" character, when appended to the mode request, specifies virtual RESERVE/RELEASE processing. Valid access modes are:

Mode Definition

- =** The user is not changing the access mode. This request only adds, deletes, or modifies access passwords.
- R** Read-only (R/O) access is desired. The access is not given if any other user has a link to the disk in write or exclusive status.
- RR** Read-only (R/O) access is desired, even if another user has a link to the disk in write status. Access is denied if any user has an exclusive link.
- SR** Stable Read-only mode is desired. (Not valid on a 370 feature system.)
- ER** Exclusive and Stable Read-only mode is desired. (Not valid on a 370 feature system.)

- W** Write access is desired. The disk is not accessible if any other user has a link to the disk in read or write status.
- WR** Write access is desired, but read-only access is acceptable if any other user has a link to the disk in read or write status. Access is denied if any user has an exclusive link.
- SW** Stable Write mode is desired. (Not valid on a 370 feature system.)
- EW** Exclusive and Stable Write mode is desired. (Not valid on a 370 feature system.)
- M** Multiple access is desired. A write link to the disk is given unless another user already has write or any exclusive access to it; in which case, the disk is not accessible.
- MR** Write access to the minidisk is desired unless another user already has write or any exclusive access to it. In this case, read-only access to the minidisk is allowed. Access is denied if any user has an exclusive link.
- MW** Unconditional write access to the disk is desired. Access is denied if any user has an exclusive or stable link.
- SM** Stable multiple write access mode is desired. (Not valid on a 370 feature system.)

readpass

if you specify a read password, then any user knowing it can establish a read-only link to your minidisk. The password must be specified as a character string of 1 to 8 characters that must not contain embedded blanks. The character string ALL permits any user to establish a read-only link without knowing the correct password.

writepass

you cannot specify this option unless you have a read password. If a user specifies a write password, then anyone else knowing it can acquire write access to the disk. The password must be specified as described in the *readpass* definition. The character string ALL is **not** recommended for use as a write access password.

multipass

you cannot specify this option unless you have both read and write passwords. If the user specifies a multiple password, then anyone else knowing it can acquire multiple-write access to the disk. The character string ALL is **not** recommended for use as a multiple-write access password.

= indicates that the specified access password is not to be changed.

DELETE

- indicates that the specified access password is to be deleted.

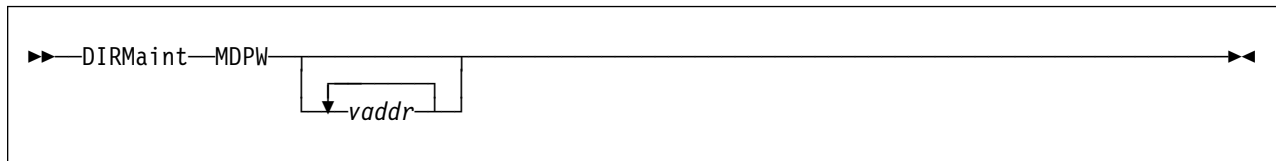
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode, PROMPT) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Use of the PROMPT prefix operand will result in command line prompting for a missing password.
3. For more information on the MDISK directory statement, see *z/VM: Planning and Administration*.
4. Most users do not want to establish a multiple-access password. You must understand the implications of establishing a multiple-access password before you do it.
5. If a password is removed, then any dependent passwords (passwords to the right of the removed password) are also removed.
6. Passwords are subject to acceptance by the user exit facility.
7. If you enter passwords directly on the command line, you should follow installation guidelines for security. It is better, from a security standpoint, not to enter access passwords on the command line. If they are omitted and the system prompts you for them, they are entered in the “display inhibit” mode. To be prompted for the passwords, specify the PROMPT prefix keyword. For example:

DIRM PROMPT MDISK 191 MR

Enter either one, two or all three passwords at the prompt.
8. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

MDPW



VM Systems: All

Purpose

The MDPW operand of the DIRMAINT command is used to obtain the access passwords for one or all of your minidisks. These passwords are part of the response message(s) written to your terminal. A message is issued for each selected minidisk, even if it does not have an access password. A header message is included in the output only if your installation is exploiting minidisk password monitoring.

Operands

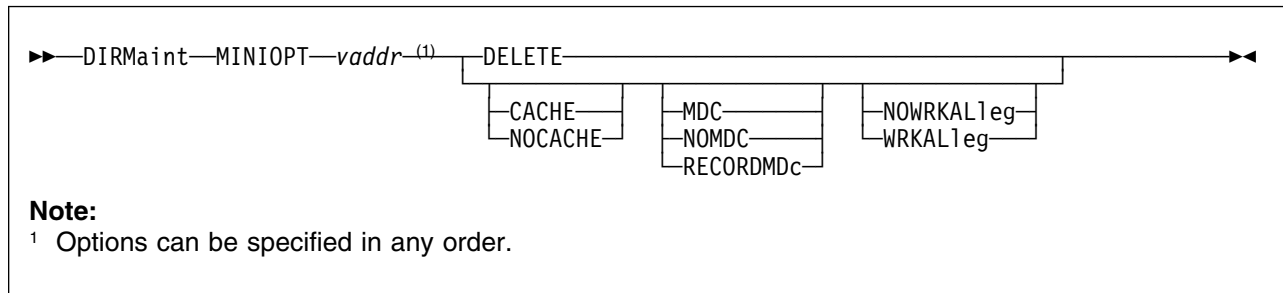
vaddr

is the virtual device address of a particular minidisk. If this parameter is not entered, all minidisks will be listed.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Your installation may have an established policy regarding passwords and security precautions. For example, many installations consider passwords to be confidential. If you are not working in a secure area, be prepared to safeguard or carefully dispose of the resulting output. If your virtual machine is in a “not receiving” state, the messages will not be recorded in your message log.

MINIOPT



VM Systems: VM/ESA (ESA Feature)

Purpose

The MINIOPT operand of the DIRMAINT command is used to add, replace, or delete an existing MINIOPT statement.

Operands

DELETE

means that the specified MINIOPT directory control statement should be deleted.

vaddr

is the virtual address of the MDISK to which the new or changed MINIOPT statement should apply.

CACHE

NOCACHE

indicates if the DASD will have access to the cache, or if CP will force I/O for the minidisk to bypass the cache.

MDC

NOMDC

RECORDMDC

indicates whether the minidisk will use full track minidisk cache, record level minidisk cache, or no cache.

Attention

The RECORDMDC option should only be used when directed to do so by IB support personnel. RECORDMDC is for use only when DASD cache as well main storage and expanded storage minidisk cache are all being over-whelmed by excessive unreferenced records being read by the full tracache support. For more information, see *z/VM: Planning and Administration*.

WRKALleg

NOWRKALleg

indicates whether working allegiance will be simulated on the minidisk.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Not every option is valid in every system environment. For more information on the MINIOPT directory statement, see *z/VM: Planning and Administration*.
3. A MINIOPT statement is a logical extension to an MDISK statement in the directory. This statement is used to specify whether CP should use the caching function for I/O to that minidisk.
4. When the DIRMAINT command is issued with the MINIOPT operand (add or replace transaction) for a device which already has an associated MINIOPT statement, the existing MINIOPT statement is replaced. The new MINIOPT statement contains only the options specified for the MINIOPT operand.

MMDISK

```
▶▶—DIRMaint—MMDisk—FROM—userid1—vaddr1—TO—userid2—vaddr2—▶▶
```

VM Systems: All

Purpose

Use the MMDISK command to copy the extents of an existing permanent minidisk in a user definition to another user definition.

The extents (device type, starting cylinder, number of cylinders, and volume serial) that define the source minidisk are copied to the target. No data movement occurs. After the MMDISK operand is processed, both the source and the target minidisk point to the same location on DASD. None of the fields of the source minidisk statement are changed. None of the fields of the target minidisk statement are changed except the extent fields. For example, the default link mode and password of this minidisk stay the same. Associated MINIOPT or DASDOPT statements are not moved.

Operands

FROM

is a required keyword

userid1

is the user ID that owns the source minidisk.

vaddr1

is the virtual device address of the source minidisk.

TO

is a required keyword.

userid2

is the user ID that owns the target minidisk.

vaddr2

is the virtual device address of the target minidisk.

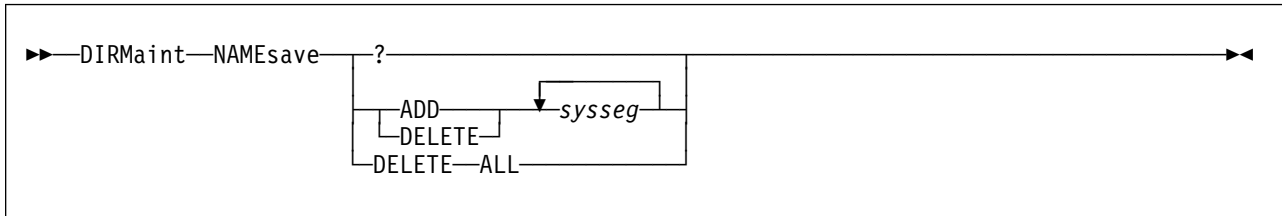
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. A profile definition cannot be specified for *userid1* or *userid2*.
3. A specification of *userid1 = userid2* when *vaddr1 = vaddr2* is prohibited.
4. Note that if there are no other minidisks in the system that occupy the same extent as that occupied by the old target minidisk, DirMaint will consider that space as available for allocation to another minidisk (through either ADD, AMDISK, CMDISK or RMDISK). Note that it has not been

formatted, even if DISK_CLEANUP= YES is specified in the CONFIG* DATADVH file.

5. A DEVNO, T-DISK or a V-DISK MDISK can be the source or the target of a MMDISK operand.
6. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

NAMESAVE



VM Systems: VM/ESA (ESA Feature)

Purpose

The NAMESAVE operand of the DIRMAINT command is used to add, delete, or query a system segment name from an existing NAMESAVE statement for a specified user's directory entry.

Operands

? requests a query of the current NAMESAVE directory entry.

ADD

specifies that the *sysseg* is to be added to an existing NAMESAVE statement, or a new statement is to be created.

DELETE

DELETE ALL

specifies that the *sysseg* is to be deleted from the NAMESAVE statement. If the **ALL** keyword is used the entire NAMESAVE statement is deleted.

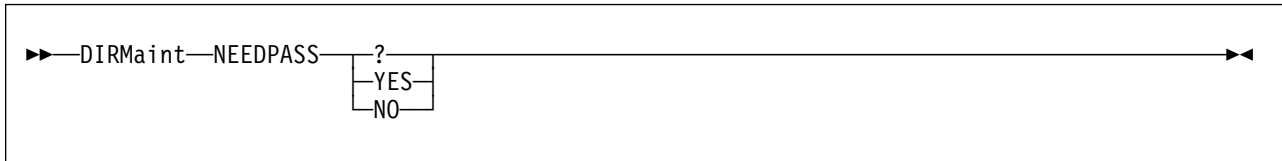
sysseg

specifies a 1 to 8 character name of the restricted named saved system, or a discontinuous saved segment that the virtual machine is authorized to access.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the NAMESAVE directory statement, see *z/VM: Planning and Administration*.
3. A "DIRM NAMESAVE ADD *sysseg*" may result in the creation of a NAMESAVE statement if there are no existing NAMESAVE statements in the directory entry, or no more room on an existing NAMESAVE statement.
4. A "DIRM NAMESAVE DELETE *sysseg*" may result in the deletion of a NAMESAVE statement if there are no more *syssegs* on the existing NAMESAVE statement.
5. Specifying DELETE for a *sysseg* name not on a NAMESAVE statement will generate an error message.

6. Specifying **ADD** for a *sysseg* name already on a **NAMESAVE** statement will also generate an error message.

NEEDPASS


VM Systems: All

Purpose

The NEEDPASS operand of the DIRMAINT command is used to activate and deactivate password prompting for DirMaint commands for the issuing user ID.

Operands

? allows you to query your current NEEDPASS setting.

YES

specifies that the user is to be prompted for a password when a command is entered.

NO

specifies that the user is not to be prompted for a password when a command is entered.

Usage Notes

1. All prefix keywords are ignored.
2. When you issue a DIRM NEEDPASS command, you will always be prompted for a password, regardless of your global variable settings or the DVHOPT entry in the directory entry settings. If a password is entered in response to the prompt, the user's global variable will be updated, regardless of whether the password is valid or not.
3. If DIRM NEEDPASS ? shows that the values of the global variable on the users machine and the DVHOPT entry in the directory entry settings differ, issue another DIRM NEEDPASS command to set them to the correct value and supply the correct password when prompted.
4. If your DIRMAINT transaction fails with message DVHREQ2286E when you are prompted to enter a password, there is a mismatch between the NEEDPASS value in your directory entry and your LASTING GLOBALV, issue another DIRM NEEDPASS NO command to set them to the correct value. However, if your error message does not end with message DVHREQ2289I, correct your error and issue another DIRM NEEDPASS NO command to set them to the correct value.

NEWS

▶▶—DIRMaint—NEWS—◀◀

VM Systems: All

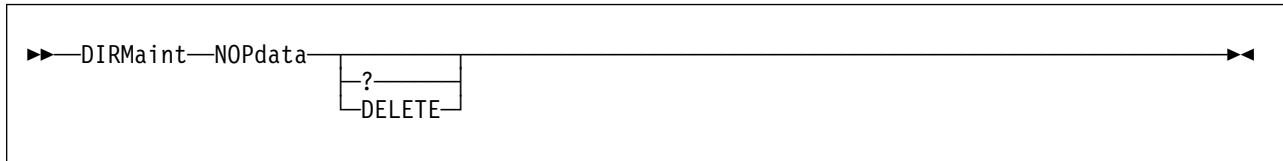
Purpose

The NEWS operand of the DIRMAINT command is used to retrieve the current news file from the DIRMAINT server.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

NOPDATA



VM Systems: VM/ESA (ESA Feature)

Purpose

The NOPDATA operand of the DIRMAINT command allows a user to add, delete, or query a NOPDATA statement in a directory entry.

Operands

? queries the directory to see if a NOPDATA statement currently exists within it.

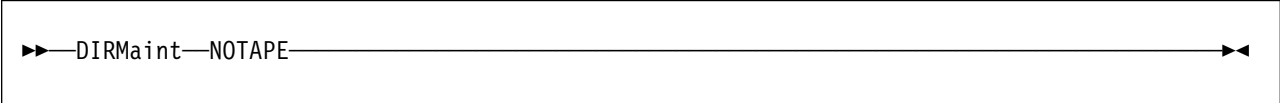
DELETE

specifies a request to delete the NOPDATA control statement from your directory entry.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the NOPDATA directory statement, see *z/VM: Planning and Administration*.
3. Requests to delete a nonexistent NOPDATA statement are rejected with a message.
4. Requests to add a NOPDATA statement are rejected with a message if the statement already exists.
5. If DELETE and ? are not specified, a NOPDATA control statement is added to the directory entry.

NOTAPE



►►—DIRMaint—NOTAPE—◄◄

VM Systems: All

Purpose

The NOTAPE operand of the DIRMAINT command is used to cancel a tape mount request issued by the backup command (BACKUP TAPE operand).

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. If your installation is using backup tapes, the BACKUP TAPE processing will issue a message to your system operator requesting that one of the tapes be mounted on a drive, and attached to the DIRMAINT virtual machine as its 181.

Thereafter, DirMaint checks for the attachment of 181 approximately every 10 minutes. If the tape is still not attached after 10 minutes, appropriate messages are issued to the operator indicating the attach has not been satisfied.

This behavior can be modified via the DVHXTAPE EXEC for interfacing with your installations tape management program.

3. This operand can be issued if the tape cannot be mounted for any reason. It cancels the mount request and the tape portion of the backup processing.

OFFLINE

OFFLINE

▶▶—DIRMaint—OFFline—————▶▶

VM Systems: All

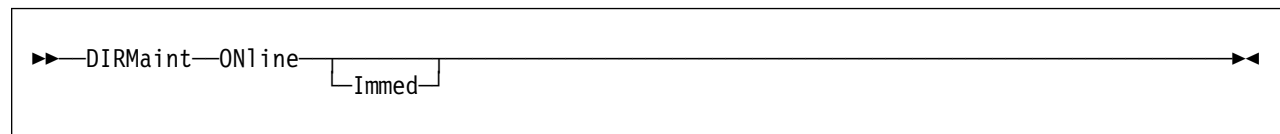
Purpose

The OFFLINE operand of the DIRMAINT command is used to temporarily suspend placing the updated source directory online.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. This operand inhibits placing the updated source directory online. Normally, the source directory is placed online as scheduled in the CONFIG* DATADVH file. The OFFLINE operand can be used to temporarily override this. This operand also inhibits CP update-in-place processing, if applicable.
3. The suspension period will last until the ONLINE operand is issued. The suspension period spans system shutdowns and restarts.
4. Manual invocation by the DIRECT operand is not inhibited in the OFFLINE state.

ONLINE



VM Systems: All

Purpose

The ONLINE operand of the DIRMAINT command is used to cancel a previous OFFLINE operand, allowing the updated source directory to be placed online.

Operands

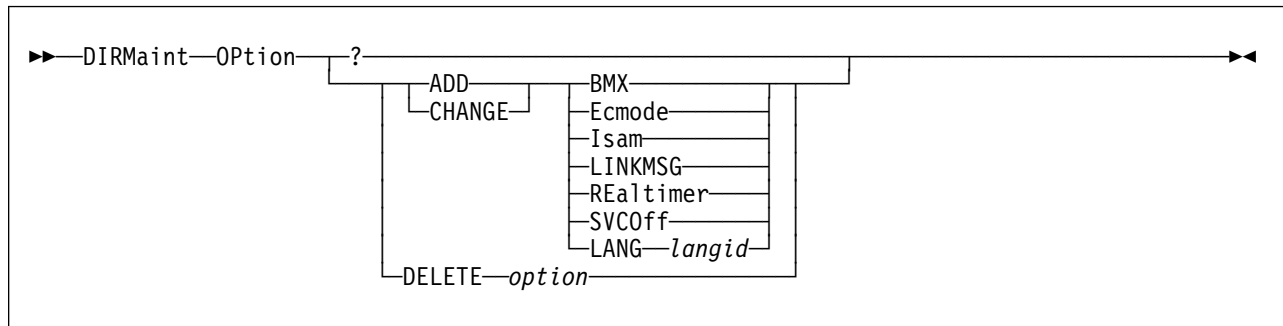
Immed

specifies that the source directory is to be placed online immediately.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. This operand cancels the effect of a previous OFFLINE operand, which prevented the placement of the updated source directory online.
3. This command causes a control file (OFFLINE CONTROL) to be deleted from the primary (and secondary if applicable) directory disk.
4. The configuration file (CONFIG* DATADVH) is consulted to determine appropriate method of placing directory changes online.
5. If you want the source directory placed online immediately, include the IMMED option with the operand. Otherwise, the directory will be placed online as scheduled in the CONFIG* DATADVH file, or by explicitly issuing the DIRECT operand.

OPTION



VM Systems: All

Purpose

The OPTION operand of the DIRMAINT command enables you to request that certain CP options be added to, changed on, or deleted from the OPTION control statement in your directory entry. Once the requested options are in your directory control statement, they will be in effect for each subsequent logon. The OPTION operand also allows you to query the OPTION control statement in your directory entry.

Note

Some options can result in a significant increase in overhead. An installation normally exercises restraint in granting use of these options to the general user.

Operands

? requests that the current OPTION statement be returned in a message.

ADD

specifies that the option that follows is to be added to the OPTION control statement in the user's directory.

CHANGE

specifies that the option that follows is to be changed on the OPTION control statement in the user's directory.

DELETE

specifies that the option that follows is to be deleted from the OPTION control statement in the user's directory.

option

specifies one of these options:

ACct

specifies that the virtual machine can issue DIAGNOSE code X'4C' to generate accounting records.

AFfinity

specifies that the virtual machine execution is done on the designated processor identified by the two-digit decimal number between 00 and 63. This operand is not valid on VM/ESA (ESA Feature) systems.

APplmon

specifies that the virtual machine can issue DIAGNOSE code X'DC'.

BMX

specifies that all virtual machine I/O operations are to occur as block multiplexer channel operations. This operand is not valid on VM/ESA (ESA Feature) systems.

CFUSER

specifies the user is authorized to connect to a Coupling Facility service machine.

CFVM

specifies the user is a Coupling Facility service machine. CFVM may not be specified with RMCHINFO, V=F, or V=R.

COMsrv

authorizes the indicated virtual machine to act as a communication server.

CONceal

places the user in a protected application at logon time.

CPuid

provides a processor identification (CPUID) in hexadecimal to be stored in response to the STIDP instruction.

DEDicate

indicates that if this user is logged on as the V=R virtual machine, every virtual processor for this virtual machine is dedicated to a real processor if possible. This operand is not valid on VM/ESA (370 Feature) systems.

DEVInfo

specifies that the virtual machine can issue DIAGNOSE code X'E4', subcodes X'00' and X'01'.

DEVMaint

specifies that the virtual machine can issue DIAGNOSE code X'E4', subcodes X'02' and X'03'.

DIAG200

specifies that the virtual machine can issue DIAGNOSE X'200'. This operand is not valid on VM/ESA (370 Feature) systems.

DIAG88

specifies that the virtual machine can issue DIAGNOSE code X'88'.

DIAG98

specifies that the virtual machine can issue DIAGNOSE code X'98'.

D84NOPAS

specifies that the virtual machine can issue all subfunctions of DIAGNOSE code X'84' except LOGPASS and MDISK, without

verification of the existing logon password. This operand is not valid on VM/ESA (370 Feature) systems.

Ecmode

lets this virtual machine run in extended control mode. This operand is not valid on VM/ESA (ESA Feature) systems.

IGNMAXU

indicates that this virtual machine can log on to the system even if the number of users already logged on is equal to or greater than the maximum allowed. This operand is not valid on VM/ESA (370 Feature) systems.

Isam

provides special channel command word translation routines that permit OS/PCP, MFT, and MVT ISAM programs to operate properly in a virtual machine. This operand is not valid on VM/ESA (ESA Feature) systems.

LANG

identifies the language that should be deleted. Used during DELETE.

LANG *langid*

identifies the language that should be set for the virtual machine during logon. Used during ADD and CHANGE.

LINKMSG

indicates a user running disconnected will receive message DMKLN116E when issuing a CP LINK command.

LKFAC

indicates that the specified virtual machine is authorized to use the multipath lock facility RPQ simulation support. This operand is not valid on VM/ESA (370 Feature) systems.

LNKExclu

specifies that the virtual machine is authorized to use the stable and exclusive access modes. This operand is not valid on VM/ESA (370 Feature) systems.

LNKNopas

specifies that the virtual machine is authorized to link to any other virtual machine's DASD without password authorization. This operand is not valid on VM/ESA (370 Feature) systems.

LNKStabl

specifies that the virtual machine is authorized to use the stable access modes. This operand is not valid on VM/ESA (370 Feature) systems.

MAIntccw

authorizes the specified virtual machine to use diagnostic CCWs. This operand is not valid on VM/ESA (370 Feature) systems.

Mlh

specifies that CP simulate an interrupt for the virtual machine whenever it detects a missing interrupt condition for an I/O operation.

NOChain

Tolerated for migration compatibility from VM/SP HPO. If specified, it will be ignored. This operand is not valid on VM/ESA (370 Feature) systems.

NODEdicate

indicates that if this user is logged on as the V=R virtual machine, no virtual processor for this virtual machine is dedicated to a real processor. This operand is not valid on VM/ESA (370 Feature) systems.

NOMdcfs

specifies that the virtual machine can use minidisk cache at a rate that is not limited by the fair share limit. This operand is not valid on VM/ESA (370 Feature) systems.

NOVf

specifies that the virtual machine may not use the Vector Facility. This operand is not valid on VM/ESA (370 Feature) systems.

Pma

allows an MVS/SP™ or VSE V=R virtual machine to use preferred machine assist or preferred machine assist with control switch assist. This operand is not valid on VM/ESA (370 Feature) systems.

Quickdsp

causes a virtual machine to be added to the dispatch list immediately when it has work to do.

REaltimer

provides a timer for the virtual machine that is updated during virtual processor runtime and during virtual wait time. This operand is not valid on VM/ESA (ESA Feature) systems.

RMchinfo

specifies that the virtual machine is authorized to access real-machine configuration information. This operand is not valid VM/ESA (370 Feature) systems; and may not be specified with CFVM.

SEorig

specifies that the virtual machine can issue DIAGNOSE X'F8', subfunction X'00'. This operand is not valid on VM/ESA (370 Feature) systems.

STFirst

specifies that the virtual machine is authorized to use the SET STBYPASS command when virtual machine assist is active on the system for a V=V user. This operand is not valid on VM/ESA (370 Feature) systems.

STGexempt

specifies that the virtual machine is exempt from CP free storage limit detection. This operand is not valid on VM/ESA (370 Feature) systems.

SVCAccl

specifies that the virtual machine can use the Supervisor Call Instruction Acceleration (SVCAccl) when IX/370 is running under CP. This operand is not valid on VM/ESA (370 Feature) systems.

SVCOff

specifies that CP, instead of the virtual machine assist feature or the VM/370 Extended Control Program Support, handles all SVC interrupts for the virtual machine.

SVC76VM

specifies that errors are not recorded by CP. This operand is not valid on VM/ESA (370 Feature) systems.

SVMstat

specifies that the virtual machine is a service virtual machine. This operand is not valid on VM/ESA (ESA Feature) systems.

Todenable

specifies that the user may change the virtual machine's time-of-day clock with the SCK instruction. This operand is not valid on VM/ESA (370 Feature) systems.

VCunoshr

specifies that all devices connected to the virtual machine are to be supported using NONSHARED protocol for virtual I/O operations. This operand is not valid on VM/ESA (ESA Feature) systems.

VIRT=FIXED

V=F

specifies that this guest is a V=F guest, if possible. This operand is not valid on VM/ESA (370 Feature) systems; and may not be special with CFVM.

VIRT=REAL

V=R

specifies that this guest is a V=R guest, if possible; and may not be specified with CFVM.

VMsave

specifies that the virtual machine contents are to be saved if CP370 is terminated or if CP370 terminates the indicated virtual machine. This operand is not valid on VM/ESA (ESA Feature) systems.

XMEM

enables MVS™ cross memory services assist for the MVS/System Product Virtual Machine. This operand is not valid on VM/ESA (370 Feature) systems.

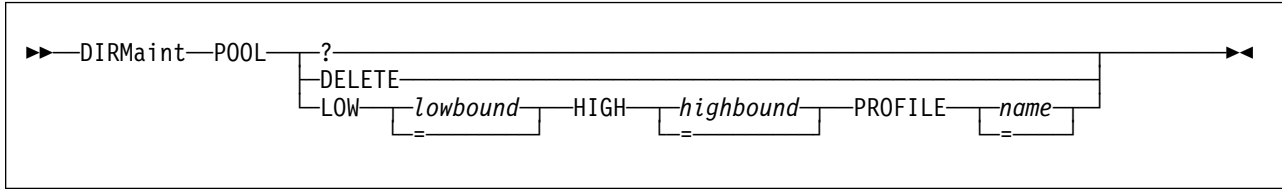
370e

specifies that the MVS/System Extensions support be enabled for the indicated virtual machine. This operand is not valid on VM/ESA (ESA Feature) systems.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOSys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. You may delete any existing values from the OPTION directory statement, however, you can only add or change those values that can be changed after logon using a CP SET, CMS SET, or other system commands. To add or change other values requires the use of the privileged SETOPTN command.
3. Not every option is valid for every release. For a complete description of the OPTION directory statement see *z/VM: Planning and Administration*.

POOL



VM Systems: All

Purpose

The POOL operand of the DIRMAINT command is used to set or alter a POOL statement following a user statement. The POOL statement allows a set of virtual machines to be defined with the same configuration or characteristics.

Operands

? requests that the current pool statement be returned in a message.

DELETE

requests that the current pool statement be deleted.

LOW

specifies that the number to follow will represent the lowest value to use when assigning user IDs.

lowbound

specifies a number from 0 to 99999 that will be used as the suffix number for the first virtual machine.

HIGH

specifies that the number to follow will represent the highest value to use when assigning user IDs.

highbound

specifies a number from *lowbound* to 99999 that will be used as the suffix number for the last virtual machine.

PROFILE

specifies that the following field will be used as the profile definition for each virtual machine in the pool.

name

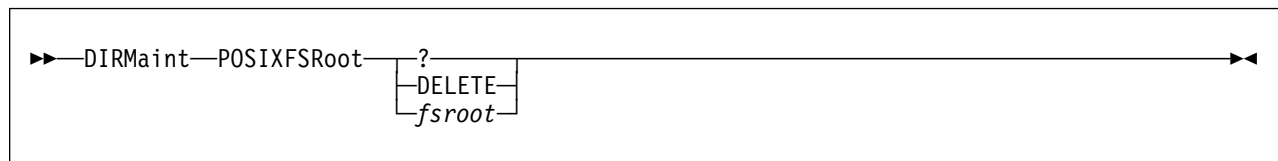
a valid profile entry in the directory.

= specifies that the existing value in the position should be maintained in the new directory statement.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the POOL directory statement, see *z/VM: Planning and Administration*.
3. The target user ID of a POOL request must be 3 characters in length or less.

POSIXFSROOT



VM Systems: VM/ESA Version 2.1.0 or later.

Purpose

The POSIXFSROOT operand of the DIRMAINT command is used to add, delete, alter, or query the FSROOT setting on a directory POSIXINFO statement.

Operands

? requests that the current settings of the FSROOT field on the POSIXINFO statement be returned in a message.

DELETE

requests that the FSROOT field of the POSIXINFO statement be deleted.

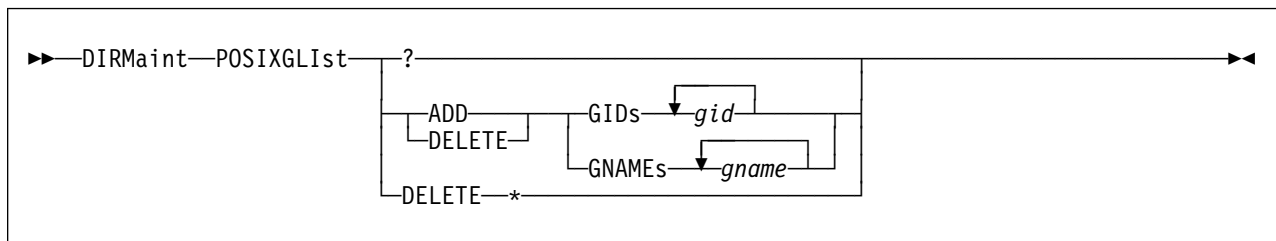
fsroot

a string placed into the POSIXINFO statement in the FSROOT field.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. To include imbedded blanks or special characters within a field requires the use of quoted strings.
3. For more information on specifying quoted strings and the POSIXINFO directory statement, see *z/VM: Planning and Administration*.

POSIXGLIST



VM Systems: VM/ESA Version 2.1.0 or later.

Purpose

The POSIXGLIST operand of the DIRMAINT command is used to add, delete, or query POSIX groups on a user's POSIXGLIST directory statement. The POSIXGLIST directory statement lists POSIX groups of which a user is a member. Each group on the list can be specified by either group ID (*gid*) or group name (*gname*). POSIXGLIST statements are the primary source for the GIDs that form the user's supplementary group list.

Operands

? returns messages containing a list of the POSIX groups the user is currently a member of.

ADD

DELETE

either adds or deletes the specified POSIX groups.

DELETE *

requests that the entire POSIXGLIST statement be deleted.

GIDs *gid*

specifies the POSIX group (by group ID) to be added or deleted in the range of 0 to 4294967295.

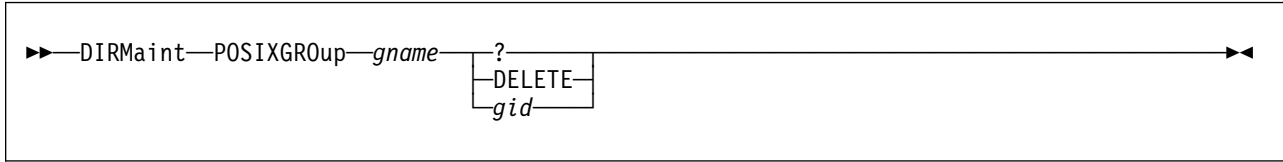
GNAMEs *gname*

specifies the POSIX group (by group name) to be added or deleted.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the POSIXGLIST directory statement, see *z/VM: Planning and Administration*.
3. If there are more than 31 GIDs and GNAMEs total on all of the POSIXGLIST statements in this directory entry, the order of them may be significant. For more information on controlling the order in which the GIDs and GNAMEs, see "SUPGLIST" on page 248.

POSIXGROUP



VM Systems: VM/ESA Version 2 Release 1.0 or later.

Purpose

The POSIXGROUP command is used to add, delete, alter, or query a POSIXGROUP statement.

Operands

gname

is the group name associated with this request.

? requests that the current settings of the specified group be returned in a message.

DELETE

requests that the specified group be deleted.

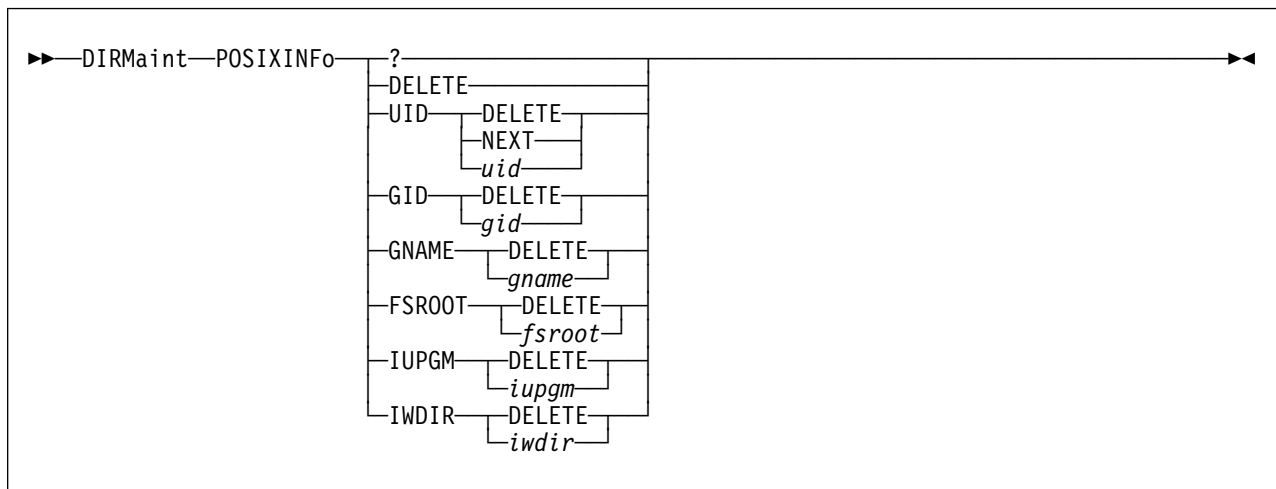
gid

a Group ID to be associated with the specified *gname*. If the group does not exist one is added. If it does, it is replaced with the new setting.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. These files will create the \$DIRGRP DIRMPART file if none exists. Also, this file contains the GLOBALDEFS statement that is used to define the GLOBALOPTS and POSIXGROUP statements.
3. For more information on the POSIXGROUP directory statement, see *z/VM: Planning and Administration*.

POSIXINFO



VM Systems: VM/ESA Version 2 Release 1.0 or later.

Purpose

The POSIXINFO operand of the DIRMAINT command is used to add, delete, alter or query the settings on a directory POSIXINFO statement.

Operands

? requests that the current settings of your POSIXINFO statement be returned in a message.

DELETE

specifies that the entire POSIXINFO statement be deleted or the listed field should be deleted. The action depends on the context in which the keyword is specified.

UID

specifies that the POSIX user ID field of the POSIXINFO statement is to be set or reset.

NEXT

requests that a UID be chosen from the POSIX_UID_AUTO_RANGE setting in the configuration file.

GID

specifies that the POSIX Group ID field of the POSIXINFO statement is to be set or reset.

GNAME

specifies that the POSIX Group Name field of the POSIXINFO statement is to be set or reset.

FSROOT

specifies that the File System Root field of the POSIXINFO statement is to be set or reset.

IUPGM

specifies that the Initial User Program field of the POSIXINFO statement is to be set or reset.

IWDIR

specifies that the Initial Working Directory field of the POSIXINFO statement is to be set or reset.

uid

a value to be placed into the POSIXINFO statement in the UID field.

gid

a value to be placed into the POSIXINFO statement in the GID field.

gname

a value to be placed into the POSIXINFO statement in the GNAME field. The GNAME is listed on a POSIXGROUP statement that has occurred near the top of the directory (before any USER or PROFILE definitions) and resolves to a GID.

fsroot

a string to be placed into the POSIXINFO statement in the FSROOT field. Quoted strings may be used (see usage note 3).

iupgm

a string to be placed into the POSIXINFO statement in the IUPGM field. Quoted strings may be used (see usage note 3).

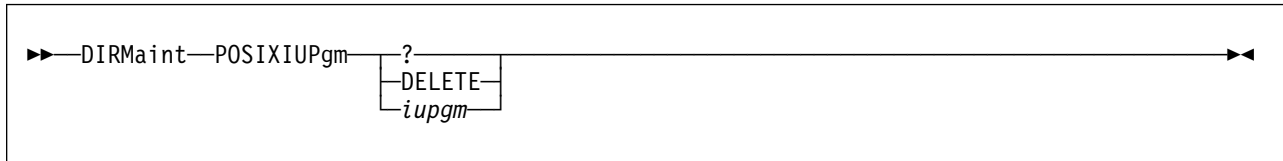
iwdir

a string to be placed into the POSIXINFO statement in the IWDIR field. Quoted strings may be used (see usage note 3).

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the POSIXINFO statement,
3. To include imbedded blanks or special characters within a field requires the use of quoted strings. For more information on specifying quoted strings, see *z/VM: Planning and Administration*.
4. Since the GNAME resolves to a GID, these two fields are mutually exclusive. Setting one field clears the other.

POSIXIUPGM



VM Systems: VM/ESA Version 2 Release 1.0 or later.

Purpose

The POSIXIUPGM operand of the DIRMAINT command is used to add, delete, alter, or query the IUPGM (Initial User Program) setting on a directory POSIXINFO statement.

Operands

? requests that the current settings of your POSIXINFO IUPGM field be returned in a message.

DELETE

requests that the IUPGM field of the POSIXINFO statement be deleted.

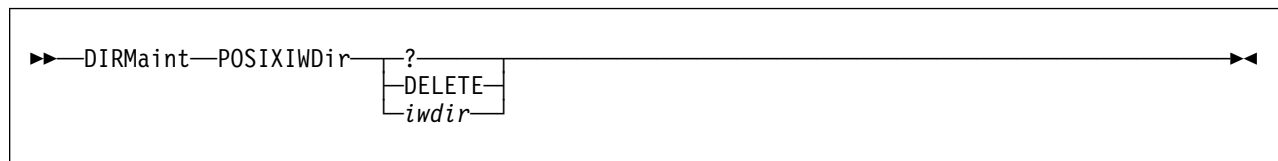
iupgm

a string placed into the POSIXINFO statement in the IUPGM field.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the POSIXINFO statement,
3. To include imbedded blanks or special characters within a field requires the use of quoted strings. For more information on specifying quoted strings, see *z/VM: Planning and Administration*.

POSIXIWDIR



VM Systems: VM/ESA Version 2 Release 1.0 or later.

Purpose

The `POSIXIWDIR` operand of the `DIRMAINT` command is used to add, delete, alter, or query the `IWDIR` (Initial Working Directory) setting on a directory `POSIXINFO` statement.

Operands

? requests that the current settings of your `POSIXINFO IWDIR` field be reported in a message.

DELETE

requests that the `IWDIR` field of the `POSIXINFO` statement be deleted.

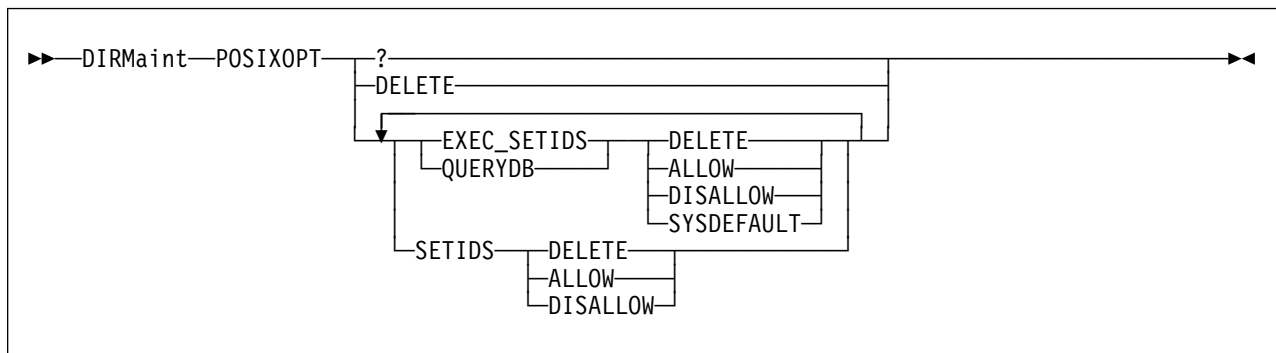
iwdir

a string placed into the `POSIXINFO` statement in the `IWDIR` field.

Usage Notes

1. You may use one or more of the prefix keywords (`REQuest`, `TOsys`, `ASuser`, `BYuser`, `FORuser`, `ATnode`) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the `POSIXINFO` directory statement, see *z/VM: Planning and Administration*.
3. To include imbedded blanks or special characters within a field requires the use of quoted strings. For more information on specifying quoted strings, see *z/VM: Planning and Administration*.

POSIXOPT



VM Systems: VM/ESA Version 2 Release 1.0 or later.

Purpose

The POSIXOPT operand of the DIRMAINT command is used to add, delete, alter, or query a POSIXOPT statement.

Operands

? requests that the current settings of the POSIXOPT statement be returned in a message.

DELETE

when specified alone (without being used in conjunction with a specific field), this operand requests that the POSIXOPT statement be deleted. When specified in conjunction with a specific field, this operand requests that the specific field being mentioned be deleted.

EXEC_SETIDS

specifies whether the user is authorized to have their POSIX security values changed on behalf of a POSIX EXEC() call designating a file with the *setuid* or *setgids* attributes.

QUERYDB

specifies whether the user is authorized to query another user's POSIX database information.

SETIDS

specifies whether the user is authorized to set another user's POSIX UIDs or GIDs.

ALLOW

DISALLOW

specifies whether the preceding specified action is to be allowed or disallowed.

SYSDEFAULT

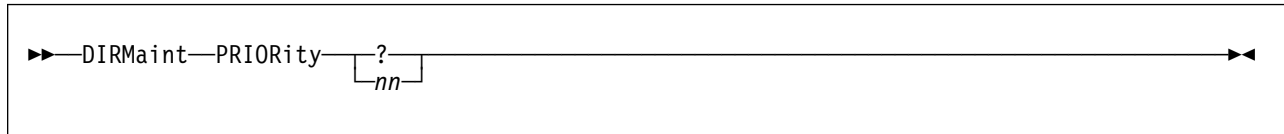
specifies the system default authorization.

POSIXOPT

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the POSIXOPT directory statement, see *z/VM: Planning and Administration*.

PRIORITY



VM Systems: All

Purpose

The PRIORITY operand of the DIRMAINT command is used to query or change the CP dispatching priority in a user's directory entry.

Operands

where:

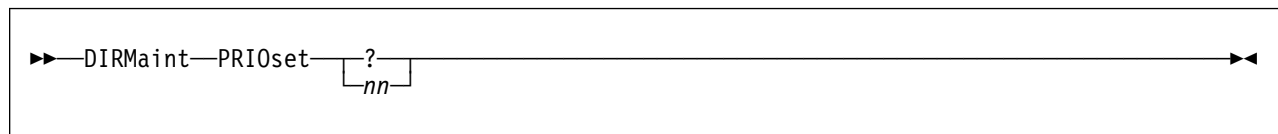
? requests that the current priority be displayed.

nn is a two-digit decimal number that is to be the new CP dispatching priority. The highest priority that can be given a user is 01, the lowest (slowest) is 99. The normal default for the scheduler is 64.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. PRIOSET and PRIORITY are synonymous with SETPRIORITY. SETPRIORITY is the preferred operand. The others are retained for compatibility with previous releases.
3. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

PRIOSET



VM Systems: All

Purpose

The PRIOSET operand of the DIRMAINT command is used to query or change the CP dispatching priority in a user's directory entry.

Operands

where:

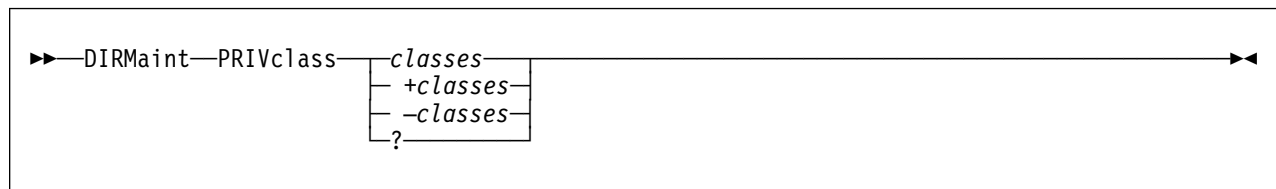
? requests that the current priority be displayed.

nn is a two-digit decimal number that is to be the new CP dispatching priority. The highest priority that can be given a user is 01, the lowest (slowest) is 99. The normal default for the scheduler is 64.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. PRIOSET and PRIORITY are synonymous with SETPRIORITY. SETPRIORITY is the preferred operand. The others are retained for compatibility with previous releases.
3. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

PRIVCLASS



VM Systems: All

Purpose

The PRIVCLASS operand of the DIRMAINT command can be used to assign, query, or delete CP privilege classes in a user's directory entry.

Operands

classes

specifies all the privilege classes which are being assigned to this user ID. Any classes currently existing, including class G, will be replaced by those specified on this command.

+classes

specifies the privilege classes to be added to this user ID. These additional privilege classes will be added to those already existing in the user directory entry.

-classes

specifies the privilege classes to be deleted from this user ID. Any other existing privilege classes will remain.

? specifies that a message should be returned listing the privilege classes currently assigned in the user directory entry.

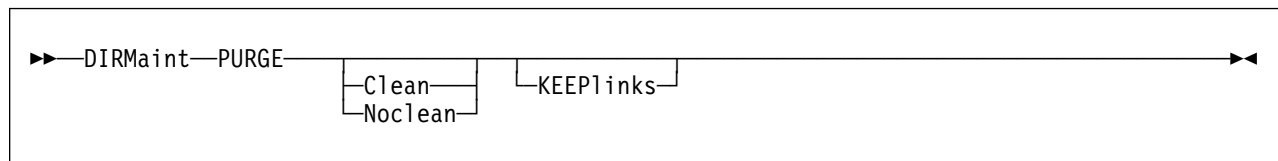
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the CLASS directory statement and how it relates to the USER statement, see *z/VM: Planning and Administration*.
3. The classes supplied with the command must be specified as a contiguous string of letters and numbers.
4. When using the *+classes* option, if any of the supplied classes are already in effect, the transaction is rejected and the user's directory entry is not updated.
5. When using the *-classes* option, if any of the supplied classes are not currently in effect, the transaction is rejected and the user's directory entry is not updated.
6. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is

PRIVCLASS

used, you are informed by a message that both the source and online directories have been updated.

PURGE



VM Systems: All

Purpose

The PURGE operand of the DIRMAINT command is used to remove the entry for a user ID or profile from the source directory.

Operands

Clean

specifies that the released minidisk extents are to be transferred to a DATAMOVE machine for formatting, even if the DISK_CLEANUP= YES option is not in effect.

Noclean

specifies that the released minidisk extents are to be made immediately available for reassignment, even if the DISK_CLEANUP= YES option is in effect. (See Usage Note 4 on page 184.)

KEEPlinks

Requests that all links be left as currently defined.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The PURGE request will be rejected in these situations. Attempts to delete:
 - A profile if any INCLUDE statements in the directory reference it. Delete the INCLUDE statements first.
 - The user ID acting as the DIRMAINT server.
 - A user ID that is defined as a DATAMOVE machine.
 - A user ID that is defined as a SATELLITE machine.
 - A user ID that is the same as the user ID issuing the command.
 - A user ID that is locked or a user ID that has device locks pending.
3. Directory entries containing MDISK statements need to be handled by the DASD subsystem asynchronously. There may be a short delay between the completion of the PURGE command and the directory entry being removed from the source directory. This is to allow the DASD subsystem to release the extents or format and release the extents.

PURGE

During this period of time commands targeting the directory will be rejected and the user's password will be set to NOLOG. A CP QUERY command issued against the user ID being purged may show that the user ID is still present. The invoker of the PURGE command will be notified when DASD processing has completed.

If cleanup is the normal mode of operation, it can be bypassed by the NOCLEAN option. If cleanup is not in effect, it can be forced for a given transaction by including the CLEAN option.

4. When you purge a user ID with full-volume minidisks that deliberately overlay other users' minidisks, do not use the CLEAN option. The default action is NOCLEAN for overlapping minidisks.

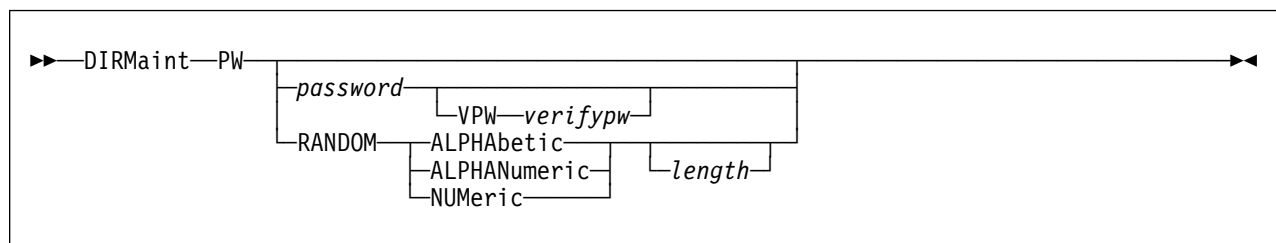
Note: If you use the CLEAN option, you will destroy those users' data.

5. Since profiles cannot contain MDISK statements, the NOCLEAN and CLEAN options are not valid when purging a profile and will be ignored.
6. PURGE processing removes any references to the user ID being purged from DirMaint control files. It also generates low priority work to remove any LINK statements that exist in other users' directory entries to minidisks owned by the user ID being deleted, unless the KEEPLINKS option is used. Purge processing also removes the subject user ID's DVHLINK FILE entries to other minidisks.

Note: Because the removal of LINK statements from other users' entries is scheduled as low priority work, it is possible that one of the users with a link statement could execute a LINK operand with a DELETE option before the low priority command executes. Another possibility is that the DLINK is unable to locate any LINK statements to be deleted. In either case, the low priority command fails since the link has already been removed. This "error" condition can be safely ignored.

7. When the PURGE operand is invoked with the CLEAN option all minidisks owned by the user are normally transferred to an available DATAMOVE virtual machine for formatting.
8. DEVNO MDISKs will not be formatted by DATAMOVE. If there is residual data that must be protected, it is the responsibility of the owning user to FORMAT the disk or otherwise obliterate the data before requesting the administrator to purge the user ID.
9. When a PURGE CLEAN is issued, any MDISK statements existing in the user ID's directory will be eliminated asynchronously through the DASD subsystem. This is done with an internally generated command called ZAPMDISK. For more information on ZAPMDISK, (Auxiliary DMDISK), see "Dasd Management" in the *Directory Maintenance VM/ESA: General Information*. If the ZAPMDISK fails, because someone else still has the link accessed (rc=104 - CP LINK error), the purge processing will not complete immediately. DATAMOVE will periodically retry the LINK and when it does succeed the ZAPMDISK will complete and the PURGE processing will continue.

PW



VM Systems: All

Purpose

The PW operand of the DIRMAINT command is used to change your CP logon password.

After entering the DIRMAINT PW command, you are normally prompted to enter your password twice.

Operands

password
specifies the value to be used as a password.

VPW *verifypw*
specifies the same password again for typographical verification.

RANDOM
requests that a random password be generated by calling the PASSWORD_RANDOM_GENERATOR_USER_EXIT.

ALPHAbetic
requests that the random generator exit be provided with the keyword ALPHABETIC in the requested algorithm field.

ALPHANumeric
requests that the random generator exit be provided with the keyword ALPHANUMERIC in the requested algorithm field.

NUMeric
requests that the random generator exit be provided with the keyword NUMERIC in the requested algorithm field.

length
requests that the random generator exit be provided with the value *length* as an optional parameter in the range 1 to 8.

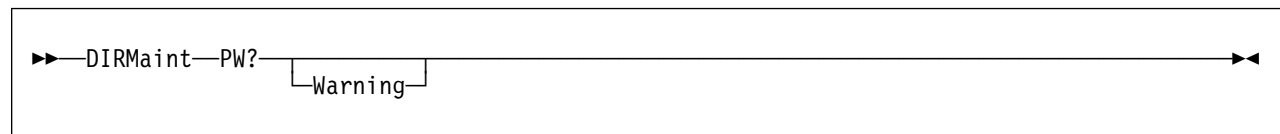
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Your installation may have rules about password content, minimum length, or reuse of old passwords. These can be enforced by calling the

PASSWORD_SYNTAX_CHECKING_USER_EXIT exit routine. For more information, see *Directory Maintenance VM/ESA: Tailoring and Administration Guide*.

3. Passwords generated by a call to the PASSWORD_RANDOM_GENERATOR_USER_EXIT exit are still passed through the PASSWORD_SYNTAX_CHECKING_USER_EXIT exit for verification.
4. Your installation may maintain a list of passwords whose use are restricted. The transaction will be rejected with an error message if your new password is on the restricted list.
5. All passwords are checked by the installations PASSWORD_SYNTAX_CHECKING_EXIT, even though they have already passed the checking by the USER exit.
6. If the verification password does not match the first password value entered, the transaction is rejected.
7. Your installation may have a password change policy and have instituted password change monitoring and enforcement facilities. Password change enforcement can be automatic or manual, and you may receive messages pertaining to password change requirements. The contents of these messages depend upon which mode is in effect.
8. You can check how many days have elapsed since you last changed your password by inserting the command EXEC DIRMAINT PW? in your PROFILE EXEC. If you append the entry with the WARN parameter, you will only be notified when your password is near expiration.
9. Your installation is able to define the maximum number of days that may elapse between password changes. When you are within a few days of the limit, you may receive reminders that a change is due. Such notification may be by direct contact from the administrative staff, internal mail, or other means. Alternatively, the installation may elect to have the DIRMAINT virtual machine contact you directly.
10. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84') if it is enabled.

PW?



VM Systems: All

Purpose

The PW? operand of the DIRMAINT command is used to determine the date of the previous CP logon password change. Pending system password changes are also indicated in the response.

Operands

Warning

specifies that a response is not returned to you unless your password is within the expiration warning period.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. On a system with an active External Security Manager (ESM) exit this command is rejected with a message since passwords are under control of the ESM.
3. The invocation of the PW? operand can be automated. The DIRMAINT command with the PW? operand can be placed in the PROFILE EXEC to automate the process as follows:

```
EXEC DIRMAINT PW?
```

Note: The full spelling of DIRMAINT is required and it must be prefixed with EXEC.

If you only want a response when password expiration is near or a system password change is pending, add the WARNING option as follows:

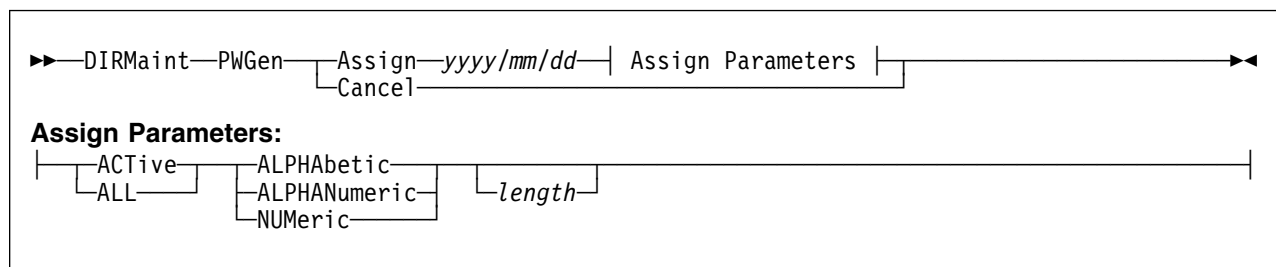
```
EXEC DIRMAINT PW? WARNING
```

4. If the DIRM PW? command is issued while an administrator scheduled password change is pending, then the user will be required to enter their current password for validation. The response supplies both the new password and the effective date.
If the DIRM PW? command is issued when no administrator scheduled change is pending, the user is not required to enter their current password. The response supplies the number of days remaining before the password must be changed by the user.
5. When the DIRMAINT PW? command is invoked, the user is informed (by a message) of the current age of their password and the number of days remaining until it is considered expired.

PW?

6. If a system password change event is pending, the user is also notified that the password will be changed. The date the change will become effective and the new password are also related in the message.

PWGEN



VM Systems: All

Purpose

The PWGEN operand of the DIRMAINT command generates a set of random CP logon passwords with the specified attributes. These passwords are then associated with the user set as specified in the command invocation.

The new passwords remain pending until the specified date is reached. A pending change may be cancelled up to the effective date by specifying CANCEL as the first and only option.

Operands

ASSIGN

causes DirMaint to generate a new password for users covered by the ALL or ACTIVE keywords.

yyyy/mm/dd

specifies the date the new passwords go into effect. In order to allow time for users to be notified, this date should be 1 to 30 days in the future.

ACTiVe

indicates that new passwords will NOT be generated for any user now having a password of AUTOONLY, LBYONLY, NOLOG, or NOPASS.

ALL

indicates that new passwords will be generated for all users.

ALPHAbetic

indicates that new passwords will consist of alphabetic characters only.

ALPHANumeric

indicates that new passwords will consist of alphabetic and numeric characters.

NUMeric

indicates that new passwords will consist of numeric characters only.

length

specifies the length of the new passwords. The required length will depend on the implementation of your random password generation exit.

Cancel

causes DirMaint to cancel any pending PWGEN operation.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Except in an emergency, it is recommended that the effective date be greater than 1 day in the future. This will allow time for users to be notified of the pending change. It also allows time to cancel the pending change if necessary.
3. Password generation requires extra processing. You may want to schedule use of the PWGEN operand for when the system is not heavily loaded. When the new passwords have been generated, a batch file called PASSWORD CHANGE is created. This file contains a list of user ID and password combinations that are due to be implemented.
4. When the passwords are created and before the passwords come into effect, the PASSWORD CHANGE file can be used for two purposes:
 - a. It can be used as input to an EXEC procedure you create to prepare notices to be sent to the affected user IDs.
 - b. It can be kept for reference and used to assist any user who was away during the password regeneration interval.

The file contains a comment as the first record (this record indicates the effective date). The remaining records are batch requests for each user to be acted on.

5. If the DIRM PW? command is issued while an administrator scheduled password change is pending, then the user will be required to enter their current password for validation. The response supplies both the new password and the effective date.

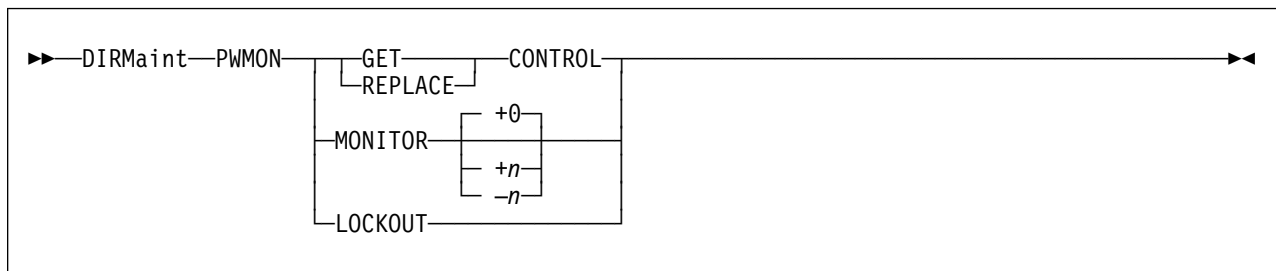
If the DIRM PW? command is issued when no administrator scheduled change is pending, the user is not required to enter their current password. The response supplies the number of days remaining before the password must be changed by the user.

6. Once the new passwords are pending, they become effective on the specified date at the scheduled backup time or at the next DirMaint IPL. When this procedure is being used, the backup time specified in the DIRMAINT DATADVH file should be as soon as possible after 00:00. This will ensure that the new passwords take effect very early on the specified date. Otherwise, any early logon users might attempt to log on before backup time and find that their old passwords are still in effect.
7. A pending PWGEN operation can be cancelled at any time between invocation and the effective date by issuing the command:

```
DIRM PWGEN CANCEL
```

If this is done, the administrator should notify all users that the expected password change has been cancelled.

PWMON



VM Systems: All

Purpose

The PWMON operand of the DIRMAINT command is used to control the CP logon password change date monitoring procedure.

Operands

GET CONTROL

requests that the current copy of the PWMON CONTROL file be sent to your virtual card reader for review (and possible revision). The PWMON CONTROL file lists user IDs that are not responsible for changing their own logon passwords, such as the the DIRMAINT virtual machine. Owners of these user IDs are also listed in this file (identified by their normal user ID) and notified when the password approaches its expiration date. This file also specifies whether the subject user ID is to be actually locked out, or is exempt from lockout with only warnings to be sent.

The PWMON CONTROL file is maintained by using an editor, all fields are delimited by one or more blanks. A general format of:

```
subject_userid YES alternate_userid alternate_node
```

or

```
subject_userid NO alternate_userid alternate_node
```

Where:

subject_userid

is the subject user ID, usually an unattended virtual machine.

YES

NO

YES specifies a lockout should occur, and NO specifies a lockout should not occur.

alternate_userid

is an alternate user ID that should receive warnings regarding the subject user ID.

alternate_node

is the node ID of the alternate user ID. If not specified, the local node is assumed.

The remainder of the line can be used for comments, such as authorization, date, and other appropriate information.

REPLACE CONTROL

requests that an original or modified PWMON CONTROL file, on one of your own accessed minidisks is to be installed in the DIRMAINT virtual machine and become effective immediately.

MONITOR

MONITOR +n

MONITOR -n

requests that the password change date monitoring procedure be executed. If MONITOR is specified alone, then date monitoring is performed with today's date as the reference point, this is the default +0. The +n (days) modifier in effect causes the reference point to be simulated as n days in the future, where as a -n modifier sets the reference point n days ago. For example, if it is Friday, then +3 simulates execution next Monday, and -4 simulates execution as it would have occurred last Monday. The overall operation of the password monitoring procedure is controlled by several statements in the CONFIG* DATADVH files, while disposition of individual cases can be influenced by the PWMON CONTROL file previously discussed.

LOCKOUT

specifies that the PWMON LOCKLIST file that was returned to you by the monitoring procedure, and modified by you (deleting lines for users who should not be actually locked out), should be sent back to the DIRMAINT virtual machine and executed. This file consists of a series of SETPW *userid* NOLOG operands that will then be executed.

The PWMON LOCKLIST file is returned to you only when the LOCKOUT mode is manual. When LOCKOUT is specified as AUTOMATIC, then the operands in this file will be executed automatically after the monitoring procedure finishes.

Usage Notes

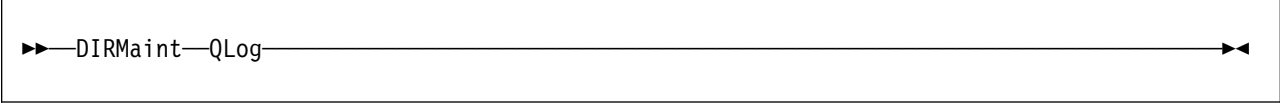
1. You may use one or more of the prefix keywords (IMMed, REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. On a system with an active External Security Manager (ESM) exit this command is rejected with a message since passwords are under control of the ESM.
3. MONITOR +n or MONITOR -n should use the MANUAL LOCKOUT method.

```
PW LOCK MODE= MANUAL
```

```
PW WARN MODE= MANUAL
```

Otherwise the the calculated cutoff date will be based upon the supplied +n or -n value.

QLOG



►►—DIRMaint—QLog—◄◄

VM Systems: All

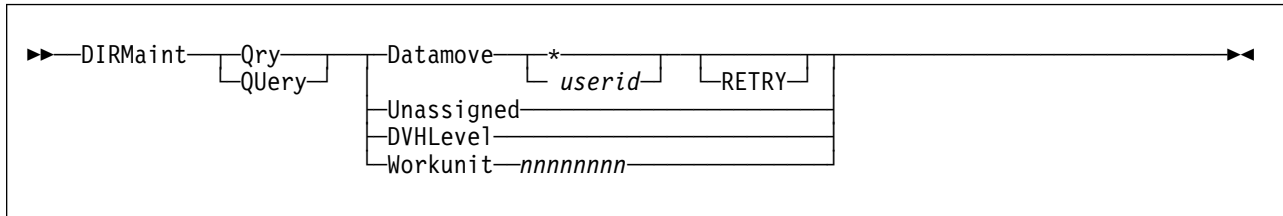
Purpose

The QLog operand of the DIRMAINT command is used to retrieve any pending messages for the originating user.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

QUERY, QRY



VM Systems: All

Purpose

The QUERY operand of the DIRMAINT command is used to request current system information from the DIRMAINT server.

Operands

Datamove

requests that the current status of one or all DATAMOVE machines be reflected.

* requests that all DATAMOVE machines be queried.

userid

requests that a specific DATAMOVE machine be queried.

RETRY

requests that the current number of pending elements and the status of the specified DATAMOVE machine be displayed.

Unassigned

requests that the status of the current unassigned workunit queue be displayed.

DVHLevel

requests that the current DirMaint system level file be displayed.

Workunit nnnnnnnn

requests that detailed information associated with the specified Work Unit Control File be shown.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

REPLACE

►►—DIRMaint—REPlace—◄◄

VM Systems: All

Purpose

The REPLACE operand of the DIRMAINT command is used to place an updated version of a user or profile directory entry in the source directory. This operand is typically used after the entry has been extracted with the GET operand and updated in the requestor's virtual machine using an editor.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Prior to invoking this operand, you can use the GET operand to retrieve the directory entry for this user ID or profile. After loading the returned file, you can modify it as necessary with the CMS editor.
3. You cannot replace a user ID with a profile, or vice versa.
4. This GET, edit, and REPLACE cycle should not be used to add a new minidisk, or to change an existing one to a new size or extent. There are specific operands for those options. Use the AMDISK operand to add a new minidisk to a user ID, or the CMDISK/RMDISK operands to change an existing minidisk in terms of size, extent, and volume residence. You can directly change the minidisk access mode and access passwords using the MDISK operand.

While it is possible to add or change minidisks using the DIRMAINT REPLACE operand, there will be no extent validation, and automatic allocation is not performed. Also, all movement of data files between moved extents will have to be coordinated manually.

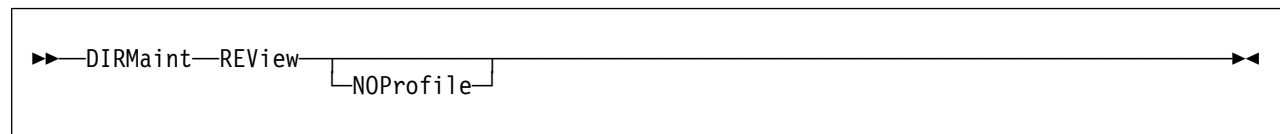
The only valid circumstance for using this method to change a minidisk is when it is to be expanded or contracted in place. For example, when a 10 cylinder minidisk is to be changed to one that is either 5 cylinders or 15 cylinders, but still is to begin at the same place on the same volume. When making this type of change, take every precaution to avoid loss of user files.

5. The DELETE option of the LINK operand is recommended when a LINK statement is to be deleted.
6. The GET, edit, and REPLACE cycle is, however, invaluable for modifying those statements in a directory entry that are not specifically supported by a DIRMAINT operand. Currently all statements are supported with the exception of comments. This method ensures that you will always have a means of performing directory changes involving any type of directory control statement.

REPLACE

7. The REPLACE operand automatically unlocks the directory entry, which has been locked since you issued the GET operand. While the entry was locked, users could not make any changes to it.
8. Any attempt to replace a user or profile definition that is not locked is rejected. This is done to minimize the possibility that this updated version would nullify an update made by the definition owner or another system administrator.
9. If the required syntax of the directory structure is violated the command will be rejected. For instance, if a USER statement were to appear inside a PROFILE definition, the REPLACE command would be rejected.
10. If an INCLUDE statement in a replacement user entry invokes a nonexistent profile, an error message will be issued and the entry will be rejected.
11. The REPLACE operand causes the generation of a new checksum which is then filed with the entry.
12. If an existing minidisk is omitted from a directory entry that is replaced, or if the allocation information for an existing minidisk is changed in a directory entry being replaced, and there is no other minidisk in the directory that occupies the same extent as that occupied by the old minidisk in the replaced entry, DirMaint will consider that space as available for allocation to another minidisk (through either ADD, AMDISK, CMDISK or RMDISK). Note that it has not been formatted, even if DISK_CLEANUP= YES is specified in the CONFIG* DATADVH file.

REVIEW



VM Systems: All

Purpose

The REVIEW operand of the DIRMAINT command allows you to review the control statements in your own directory entry. This allows you to check your virtual machine configuration and options, but not your passwords.

Operands

NOProfile

specifies that if your directory entry includes a profile, you do not want to see it expanded.

Normally the contents of the profile are included inline, set off by the following two lines:

```
DVHRXV3355I The following records are included from
profile: "name"
```

```
DVHRXV3355I The preceding records are included from
profile: "name"
```

A profile is a group of control statements that are common to several directory entries. One INCLUDE statement invokes an entire profile, just as if every control statement in the profile were coded in your directory entry. Statements in your directory entry override statements in the profile.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The logon password for the subject user ID and the password field on all APPCPASS statements are always masked with X's in the REVIEW output. In general, all minidisk access passwords will also be masked in the same fashion. Minidisks with access passwords of ALL will be left intact in the REVIEW output. This is done to highlight that there is some type of universal access to these minidisks.
3. The list of all links to minidisks owned by the subject user ID may not be complete if the installation has identified one or more of the minidisks as candidates to be excluded. The data that is referenced when generating this section of the output comes from a file other than the source directory. The content of this file is determined by the installation. If known links appear to be missing from the REVIEW output, the system administrator should be contacted. It is likely that the minidisk in question is the subject of an entry in the DVHLINK EXCLUDE file. For more information, see "USEROPTN" on page 264.

RLDCODE

►►—DIRMaint—RLDCode—◄◄

VM Systems: All

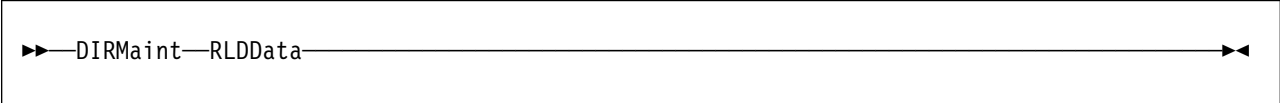
Purpose

The RLDCODE operand of the DIRMAINT command causes the DIRMAINT virtual machine to reload its resident operating procedures. The processing invoked by the RLDCODE operand is also invoked automatically as an integral part of the system initialization by means of IPL.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The DIRMAINT virtual machine runs with most of its performance critical parts preloaded into virtual storage using either EXECLOAD (for EXECs, REXX pipeline stages, and XEDIT macros) or NUCXLOAD (for MODULE files).
3. If the FILE operand is used to replace a preloaded part, the FILE operation should be followed with this RLDCODE operand to put the revised version into operation. Otherwise, it will not be available for use until the next system restart.

RLDDATA



►►—DIRMaint—RLDData—◄◄

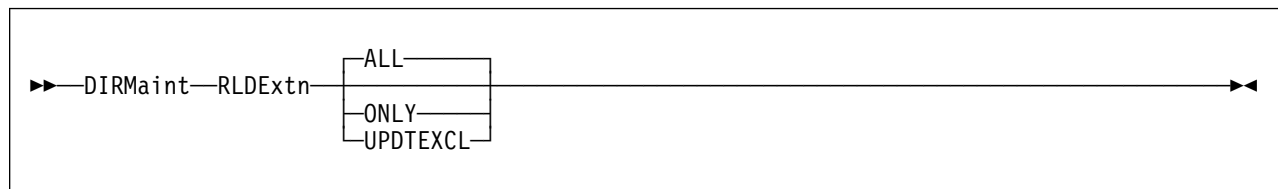
VM Systems: All

Purpose

The RLDDATA operand of the DIRMAINT command tells the DIRMAINT virtual machine to reload the data tables from the CONFIG* DATADVH disk files. The processing invoked by the RLDDATA operand is also invoked automatically as an integral part of system initialization during IPL. The RLDDATA operand does not automatically invoke the RLDEXTN function.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The DIRMAINT virtual machine loads the virtual storage data tables at IPL time from the CONFIG* DATADVH disk files. The CONFIG* DATADVH disk files contain the basic installation options and parameters.
3. The RLDDATA operand must not be used to reload a CONFIG* DATADVH file with a DATAMOVE statement that contains a different user ID than that of the currently existing DATAMOVE virtual machine. To change the name of the DATAMOVE virtual machine:
 - a. Insure that the DATAMOVE machine you intend to delete is not busy. This can be accomplished by using the DIRM QUERY command or by looking at the DATAMOVE CONTROL file on the DirMaint 1DF disk.
 - b. Stop the DIRMAINT server.
 - c. Change the CONFIG* DATADVH file
 - d. Initialize the DirMaint system with the DVHBEGIN command.
4. The EXTENT CONTROL file carries specifications for using minidisk volumes. Modifications to the EXTENT CONTROL file must be activated using the RLDEXTN operand. The RLDDATA operand no longer automatically invokes the RLDEXTN function as it did in DirMaint Release 4.0

RLDEXTN


VM Systems: All

Purpose

The RLDEXTN operand of the DIRMAINT command is used to reload the information in the EXTENT CONTROL file. It also provides a method for rebuilding the various volume control files on the system.

Operands

ALL

All functions are performed. This is the default action.

Functions include:

1. Process the Extent Control file and form all the various internal use files from the extent control file.
2. Rebuild the DASD volume control files from the source directory.
3. Rebuild the link control file from the source directory.

ONLY

Only the extent control file is processed. This option does not rebuild the volume control and link files from the source directory. This option is appropriate when the EXCLUDE section of the Extent Control file was not altered. Placing any new excluded extents into the current volume control file will not be done if this option is used.

UPDTEXCL

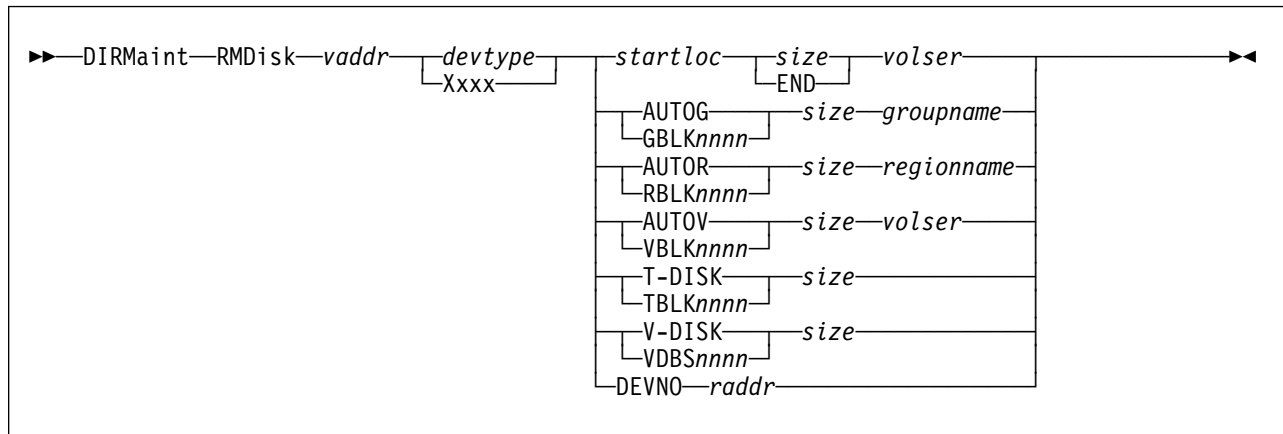
This option processes the extent control file and also manipulates the existing volume control files to reflect the new excluded extents. This function differs from ALL in that the source directory is not consulted. This may prove to be useful for very large installations.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The EXTENT CONTROL file carries specifications about how minidisk volumes are used. Its contents control the processing in the automatic minidisk allocation routines.
3. The RLDEXTN operand can be used to activate an EXTENT CONTROL file that has been modified or to force a rebuild of your volume control file structures.

4. Note that one side effect of using the UPDTEXCL option is that this method does not rebuild the control files from the source directory. Under most circumstances this is not a problem. One case where it could be a problem would occur if your site allows administrators to directly manipulate the directory source (without using DirMaint). The possibility exists that MDISK/LINK statements could be altered without the new values being reflected in DirMaint's internal files (the volume control files and the link file). Should this occur the volume control files should be rebuilt. Using the ALL option has the effect of synchronizing DirMaint's internal structure with the current directory and should be used if there is any possibility of direct source directory manipulation.

RMDISK



VM Systems: All

Purpose

Use the RMDISK operand of the DIRMAINT command to redefine an existing permanent minidisk in a user definition.

Operands

vaddr

is the virtual device address of the minidisk to be redefined.

devtype

is the device type of the volume to which the minidisk is assigned. Otherwise, if automatic allocation is used, specify the device type as XXXX. Automatic allocation options include:

- AUTOG and GBLK
- AUTOR and RBLK
- AUTOV and VBLK

operations. The real device type will be taken from the REGIONS section in the EXTENT CONTROL file for that volser.

startloc

is a decimal cylinder relocation factor or a decimal FB-512 device block number that specifies the beginning of the minidisk extent corresponding to cylinder 0 or block 0 of the user's minidisk.

AUTOG

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a volume within a specified group. The group must have been defined in the EXTENT CONTROL file, and the regions within the group must also be defined.

GBLKnnnn

is functionally equivalent to AUTOG except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: GBLK512, GBLK0512, GBLK1024, GBLK2048, GBLK4096, GBLK1K, GBLK2K, and GBLK4K. In addition,

GBLK800 or GBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

The disk will not be formatted for CMS use by DirMaint unless the BLKSIZE or LABEL operands are used, but the number of CMS blocks of a given size per cylinder is used to calculate the number of CKD cylinders or FB-512 blocks necessary to satisfy the request.

AUTOR

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a specified region or volume. The region must have been defined in the EXTENT CONTROL file.

RBLKnnnn

is functionally equivalent to AUTOR except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: RBLK512, RBLK0512, RBLK1024, RBLK2048, RBLK4096, RBLK1K, RBLK2K, and RBLK4K. In addition, RBLK800 or RBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

AUTOV

specifies that a starting CKD cylinder or FBA block number is to be automatically assigned at a suitable place on a specified volume. The volume must have been defined in the EXTENT CONTROL file.

VBLKnnnn

is functionally equivalent to AUTOV except that the size is specified in units of CMS 512, 800, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: VBLK512, VBLK0512, VBLK1024, VBLK2048, VBLK4096, VBLK1K, VBLK2K, and VBLK4K. In addition, VBLK800 or VBLK0800 may be used for device types of 3330, 3340, 3350, 3375, and 3380.

The disk will not be formatted for CMS use by DirMaint unless the BLKSIZE or LABEL operands are used, but the number of CMS blocks of a given size per cylinder is used to calculate the number of CKD cylinders or FB-512 blocks necessary to satisfy the request.

T-DISK

provides temporary disk space to a user upon logging on. This space is made available from preallocated system disk space. The owner must initialize or format this minidisk at each logon. It will be a part of the owner's virtual configuration until logoff or disk detachment, at which time the owner loses all files and the space is returned for reallocation to another user. If concerned about security, the owner should initialize or reformat the minidisk before releasing it.

TBLKnnnn

is functionally equivalent to T-DISK except that the size is specified in units of CMS 512, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: TBLK512, TBLK0512, TBLK1024, TBLK2048, TBLK4096, TBLK1K, TBLK2K, and TBLK4K.

V-DISK

provides virtual disk space to a user upon logging on. The V-DISK may already exist when the user links to it, providing other users that previously linked to it are still linked when the user logs on. Otherwise, the owner

must initialize or format this minidisk at each logon. It remains a part of the owner's virtual configuration until logoff or disk detachment. When the last linked user logs off or detaches the V-DISK, the file space is returned for reallocation to another user.

VDBS*nnnn*

is functionally equivalent to V-DISK except that the size is specified in units of CMS 512, 1024, 2048, or 4096 blocks rather than in CKD cylinders or FBA blocks. Valid values are: VDBS512, VDBS0512, VDBS1024, VDBS2048, VDBS4096, VDBS1K, VDBS2K, and VDBS4K.

size

is a 1 to 8 digit decimal number specifying the size of the new minidisk in CKD cylinders or FBA blocks. For allocation requests involving block sizes, allocation will be converted to the exact number of FB-512 blocks for FBA devices or to the minimum number of cylinders needed to contain the specified number of CMS blocks for CKD devices.

END

specifies that the minidisk will extend from the assigned starting cylinder to the end (last cylinder) of the volume in which it is defined. This parameter is valid for fixed allocation only.

groupname

specifies the name of the DASD allocation group from which DirMaint will pick the *volser* and starting CKD cylinder or FBA block. Groups consist of one or more specific *regions*. This is valid for AUTOG and GBLKnnnn requests only. Groups must be defined in the EXTENT CONTROL file, and can be 1 to 8 alphanumeric characters.

regionname

is the name associated with the *region*, 1 to 8 alphanumeric characters, on which the minidisk is to be allocated. This is valid for AUTOR and RBLKnnnn requests. The named region must be defined in the REGIONS section of the EXTENT CONTROL file. The region defines a segment on a specific DASD volume for allocation.

valid

is the volume label of the DASD volume, 1 to 6 alphanumeric characters, on which the minidisk is to be allocated, or the reserved identifier of *&SYSRES*. This applies to fixed allocation (where a specific starting cylinder or block number is specified) and to automatic allocation on a volume (AUTOV, VBLKnnnn).

DEVNO *raddr*

specifies a full volume minidisk. The use of this option is valid only in VM/ESA (ESA Feature) environment. *raddr* is the real device number of the full volume minidisk.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

2. It is not advisable to define a minidisk starting at real cylinder 0 unless it is to be used by OS ISAM. If it is to be used by OS ISAM, it must begin at real cylinder 0. If you do assign a minidisk beginning at real cylinder 0, the user who owns it must be aware that the minidisk label is the real label that the user and the system use to identify the disk. CP-owned volumes must not have minidisks beginning at real cylinder 0. Generally, CKD DASD extents should begin with cylinder 1 or higher, and FBA DASD extents should begin with block 32 or higher.
3. If there is no other minidisk in the directory that occupies the same extent as that occupied by the old minidisk in the replaced entry, DirMaint will consider that space as available for allocation to another minidisk (through either ADD, AMDISK, or CMDISK). Note that it has NOT been formatted, even if DISK_CLEANUP= YES is specified in the CONFIG* DATADVH file.
4. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

Examples

1. The following DIRMAINT RMDISK command redefines the 192 minidisk owned by SMITH (SMITH's 192 is redefined as a 9345 device starting at address 438 and extending for 6 cylinders on disk volume id VOL122):

```
DIRM FOR SMITH RMD 192 9345 438 6 VOL122
```

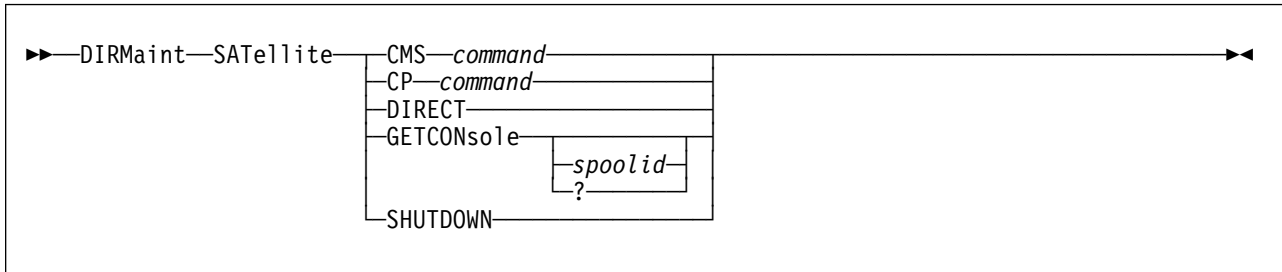
Any data previously existing in SMITH's previously defined minidisk space remains unchanged. In addition, data previously existing on the redefined minidisk also remains unchanged.

2. The following DIRMAINT RMDISK command tells DirMaint to change DAVE's existing 123 minidisk, whatever its existing device type and location, to be the real 9345 DASD unit on address 1CB:

```
DIRM FOR DAVE RMD 123 9345 DEVNO 1CB
```

The data on DAVE's existing 123 disk is not changed, but the space may become available for future allocation. The data on device 1CB is not changed. If the volume on 1CB is defined in the EXTENT CONTROL file, the otherwise unused space remains allocatable.

SATELLITE



VM Systems: All

Purpose

The SATELLITE operand of the DIRMAINT command is used to control the satellite server machines.

Operands

CMS *command*

CP *command*

executes the specified CMS or CP command on the satellite server.

DIRECT

requests that the satellite server obtain the current source directory and build a source directory.

GETCONsole *spoolid*

specifies that a copy of the identified console spool file is to be sent to the command issuer. If *spoolid* is omitted, the currently active console spool file is closed and a copy is sent to the command issuer.

Note: Specifying the spool ID of the currently active console spool file will result in an error. To obtain a copy of the currently active console spool file, omit the *spoolid* parameter.

GETCONsole ?

specifies that a list of the available console spool files is displayed to the command issuer.

SHUTDOWN

requests that the satellite server shutdown.

Usage Notes

1. You may use one or more of the prefix keywords (IMMed, REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.

SCAN

►►—DIRMaint—SCAN—*pattern*—————◄◄

VM Systems: All

Purpose

The SCAN operand of the DIRMAINT command is used to perform a high speed search of the source directory. A file with all records that match the specified pattern is returned to the requester.

Operand

pattern
defines the search pattern.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The SCAN operand is designed to identify matching directory statements. An asterisk (*) can be used alone as a wild card indicating that all values found in that position in the record are matches.
3. If the first parameter is qualified by a suffix asterisk, for example, *xxx** and is a parameter abbreviation, then a match on the corresponding field in each directory statement is unconditional.
4. The search parameter list provided to SCAN is uppcased before the search begins. This must be considered when attempting to scan for directory statements that allow mixed case arguments such as POSIXGROUP and POSIXINFO statements.
5. If the first parameter is qualified by a suffix asterisk, for example if the form *xxx**, the *xxx* can not be the full statement name. Otherwise the search will not render the desired results. The use of the asterisk implies that the preceding is an abbreviation of the full statement name. For example, DIRM SCAN MDISK* will not process a scan for the MDISK statements. DIRM SCAN MDISK will process the MDISK statement as well as DIRM SCAN MDIS*.
6. Any pattern parameter consisting only of wild card designators in the form "*****," is treated the same as a parameter consisting of a single asterisk.
7. A scan pattern consisting of nothing but wild card designators in the form
DIRM SCAN * * * * *
is equivalent to the command
DIRM SCAN *
8. The maximum number of scan arguments that are allowed in the pattern is 13.

SCAN

9. The command "DIRM SCAN *" returns all comment records within the source directory, not all records.
10. Using this operand to scan the source directory may cause sensitive information, logon and minidisk passwords, to be sent to your user ID.
11. Using pipeline control characters, for example, a vertical bar, (X'4F'), or a question mark, (X'6F'), will cause the DirMaint service machine to issue error messages, DMSSCC2827E, DMSPRS2828I, and DVHxxx3209E.

SCAN Information and Examples

The pattern to be matched should be constructed like the directory entry the user is searching for. The pattern supplied must be the positional equivalent of the directory control statement format the search is targeted for. If you want to search for any users with a maximum logon storage of 16MB you must enter the SCAN operand as follows, because the maximum storage field is the fifth field on the USER statement.

```
DIRM SCAN * * * * 16M
```

The following example will search for all user IDs with a maximum storage value of 16MB. The "*" wild card character is not limited to an entire token. It can also be used in conjunction with character strings.

```
DIRM SCAN * * * * 16M
```

```
Userid: <=== Qualifying Record =====>
ALBAHAE  USER ALBAHAE  password 2M 16M G
ALICEK   USER ALICEK   password 2M 16M G
AMD      USER AMD        password 2M 16M G
ANGELA   USER ANGELA    password 2M 16M G
ANJAVI   USER ANJAVI    password 2M 16M G
APPINO   USER APPINO    password 2M 16M G
AP2SVP   USER AP2SVP    password 2M 16M G
ATORRES  USER ATORRES   password 2M 16M G
AUSSIM   USER AUSSIM    password 2M 16M G
BAKIS    USER BAKIS     password 2M 16M G
BALU     USER BALU      password 2M 16M G
BARZILA  USER BARZILA   password 4M 16M G
```

The following example shows a source directory scan based on account number:

```
DIRM SCAN ACCOUNT 4736819
Userid: <=== Qualifying Record ===>
BAP ACCOUNT 4736819 BAP
DENISTO ACCOUNT 4736819 WAYNE
GOODWIN ACCOUNT 4736819 GODWIN
JCOYLE ACCOUNT 4736819
MAINTSY ACCOUNT 4736819 MAINTSY
NAMESRV ACCOUNT 4736819
NSMITH ACCOUNT 4736819 NSMITH
PEDERSN ACCOUNT 4736819 PEDERSN
RLM ACCOUNT 4736819 RLM
ROONEY ACCOUNT 4736819 ROONEY
RUBAS ACCOUNT 4736819 4-032
SDFTST1 ACCOUNT 4736819
SDFTST2 ACCOUNT 4736819
SVM ACCOUNT 4736819 SVM
TCPDBA ACCOUNT 4736819
TESTPRIV ACCOUNT 4736819 TESTPRIV
WIJNEN ACCOUNT 4736819
```

The * character can be used as a wild card indicating that all values found in that position in the record are matches. For example, to search for all ACCOUNT records with an account number that includes the string DEF, the following command can be issued:

```
DIRM SCAN ACC* *DEF*
```

This command will find all records that contain "ACC" as the first three characters of the first token in the record and which contain the string "DEF" anywhere in the second token in the record. Thus it will find any of the following records:

```
ACC      ABCDEF  DISTCODE
ACC      DEFGHI  DISTCODE
ACCOU    BCDEFGHI DISTCODE
ACCOUNT  DEF      DISTCODE
```

It will NOT find the following records:

```
AC      ABCDEF DISTCODE
ACC     ABC    DEF
ACCOU   ABC    DISTCODE ABCDEF
ACCOUNT ABC    DISTCODE DEF
```

The key to locating the information you are looking for in the source directory is the selection of the scan pattern to be matched. In general, pattern matching occurs in a manner analogous to that performed by the FILELIST command.

Using a wild card designator to obtain a match on specific keyword may not yield the expected results. For example, the minimum abbreviation for the IPL statement is "I," but it may also be represented by "IP" and "IPL." If you are interested in obtaining a list of what all users in the source directory are IPLing, the selection of a concise scan pattern is not obvious. One possibility is to formulate the scan pattern as follows:

```
DIRM SCAn I*
```

This method will certainly ensure that all IPL statements are found, but it has a drawback in that it will also find all INCLUDE and IUCV statements. Based on the configuration of your source directory, this may cause the output to contain far more information than you wished to see and obscure the information you are interested in. The only way to eliminate extraneous information and be sure that all possible abbreviations are accounted for is to avoid the use of wild card designators and issue three SCAN requests, one for each possibility.

To compensate for this flexibility in the way that directory statements can be specified, the SCAN processor treats the first pattern parameter in a special manner. The following describes the behavior of SCAN processing relative to the first pattern parameter:

1. If the first parameter does not contain an asterisk, that is, if it is not in the form "xxxx*", "*xxx" or "xx*xx" it is assumed to be the minimum abbreviation of the directory statement in which the invoker is interested. So for the example (DIRM SCAN I*) discussed above, the scan pattern that should be supplied to obtain the IPL statements from all users' directory definitions is:

```
DIRM SCAN I
```

The output from this invocation will contain all IPL statements, regardless of the abbreviation used in the actual statement. The minimum abbreviation for various directory statements can be found in the DIRECTXA DATADVH file (for an ESA feature system) or the DIRECT DATADVH file (for a 370 feature system) on the DIRMAINT server.

During processing of the source directory, DirMaint expands each statement keyword to its full length as it is read. This ensures that regardless of the minimum abbreviation provided, the scan output will contain the records being scanned for. For example, consider the following segment from a directory:

```
USER USER1 SECPASS1
  IN COMMON
USER USER2 SECPASS2
  INC COMMON
USER USER3 SECPASS3
  INCL COMMON
USER USER4 SECPASS4
  INCLUDE COMMON
```

Scan will find all of the common records indicated above, regardless of abbreviation, with any of the following command invocations:

```
DIRM SCAN IN
DIRM SCAN INC
DIRM SCAN INCL
DIRM SCAN INCLU
DIRM SCAN INCLUD
DIRM SCAN INCLUDE
```

This does have implications when using a leading asterisk on the first token. The pattern matching is applied against the expanded keyword regardless of the abbreviation found in the source directory.

2. If the first parameter is qualified by a suffix asterisk, (that is, of the form "xxx*"), the value comparison is performed between the parameter and the

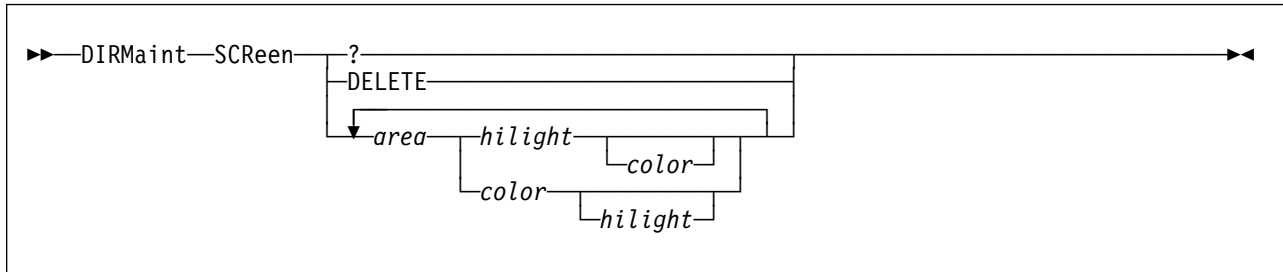
first field on the directory record for the length of the parameter character string up to, but not including, the trailing asterisk. If the first field on the directory record represents a valid abbreviation of a directory statement, the truncated parameter value is compared to the FULL statement name for the appropriate length much like the way FILELIST evaluates wild card characters.

The implications of this are best explained by the following example. Suppose that the submitted command was:

```
DIRM SCAN M*
```

SCAN processing would return all MACHINE, MDISK, and MINIOPT records in the source directory, since all of these statement types begin with the necessary "character string."

SCREEN



VM Systems: All

Purpose

The SCREEN operand of the DIRMAINT command is used to assign or change color and extended highlight definition to areas of your virtual machine console display screen. For example, the input, output, and status areas can be assigned specific colors for ease of identification to enhance screen readability. These areas are identified under *area* in the option specifications.

Operands

? requests a query of the current color and extended highlight definitions for the *area* of the screen.

DELETE

requests that the current color and extended highlight definitions for the *area* of the screen be deleted.

area

specifies the area of the screen. *area* can be:

- | | |
|----------|---|
| ALL | the entire screen. |
| INArea | the input area. |
| STAtarea | the system status area. |
| OUTarea | the output area (CPOut, VMOut, and INRedisp combined). |
| CPOut | is the output from the control program. |
| VMOut | is the output from CMS or the virtual operating system running in your virtual machine. |
| INRedisp | is the input redisplay. |

highlight

specifies the extended highlight value you want to assign to an *area* of your screen. *highlight* can be:

- | | |
|----------|-----------------|
| BLInk | blinking |
| REVvideo | reverse video |
| UNDerlin | underlining |
| NONE | no highlighting |

color

specifies the color attribute you want to assign to an *area* of your screen. The *color* can be:

| | |
|----------|--|
| BLUe | blue |
| RED | red |
| GREen | green |
| YELLow | yellow |
| TURquois | turquoise |
| PINK | pink |
| WHIte | white |
| DEFault | monochrome (green and white) is the default. |

Usage Notes

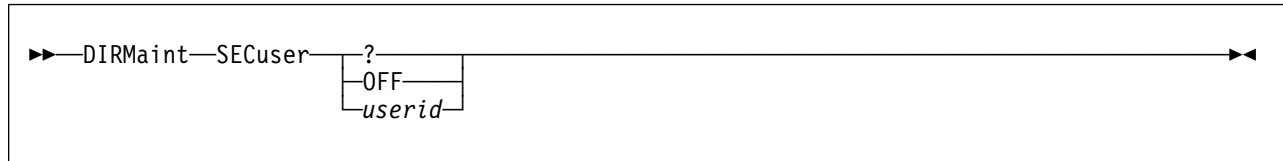
1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the SCREEN directory statement, see *z/VM: Planning and Administration*.
3. Each time you specify the SCREEN operand, you must specify at least one screen *area* option with at least one *color* or *highlight* attribute on the same command line.
4. The operands are processed from left to right and are additive. For example:


```
DIRM SCREEN ALL RED VMO BLUE
```

 would result in all areas being mapped to RED with the exception of the VMOOut area which would be mapped to BLUE.
5. The redundancy rules prohibit specifying an *area* more than once for a specific attribute type. For example, the following specification of the SCREEN operand would be rejected as redundant:


```
DIRM SCREEN ALL RED ALL BLUE
```
6. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

SECUSER



VM Systems: All

Purpose

The SECUSER operand of the DIRMAINT command allows the general user to manipulate the secondary console field on their console statement.

Operands

? queries the directory to see if a secondary user ID is currently specified on their CONSOLE statement.

OFF

specifies a request to delete the secondary user ID from their console statement.

userid

specifies the user ID to place on the CONSOLE statement to act as the secondary console.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the CONSOLE directory statement, see *z/VM: Planning and Administration*.
3. The use of SECUSER, AUTHSCIF, DSECUSER and DROPSCIF commands are closely related. The *userid* must have granted the issuing user authority through the AUTHSCIF command to use them as a secondary console. An error message will result if this has not been done. For example, if the *userid* WATCHER chooses to become the secondary console of the disconnected user SERVER, this sequence of commands is needed:
 - a. WATCHER issues DIRM AUTHSCIF SERVER
 - b. SERVER issues DIRM SECUSER WATCHER

When the secondary user facility is no longer needed, one of two commands would be needed:

- a. WATCHER issues DIRM DSECUSER SERVER
- b. SERVER issues DIRM SECUSER OFF

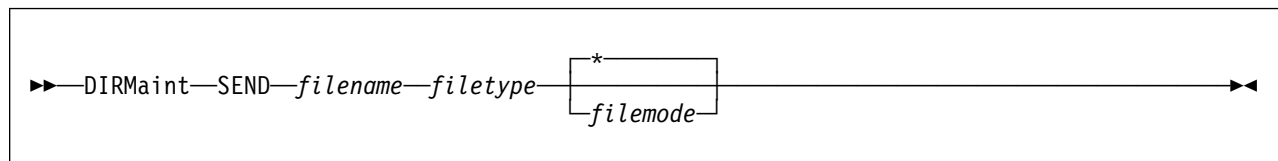
If SERVER is not going to use WATCHER as a SCIF user again, the general authority can be cancelled if WATCHER issues:

DIRM DROPSCIF SERVER

The design allows the SCIF target *userid* (WATCHER) to have overall control of the usage of SCIF, in case of SERVER produced SCIF output which the target was not able to handle. SERVER has sole control of the time it is turned on, but WATCHER must authorize it first. If the SECUSER command is issued before the nominated *userid* has granted authority with the AUTHSCIF command, an error message will result.

SEND

SEND



VM Systems: All

Purpose

The SEND operand of the DIRMAINT command is used to request that a copy of a file on one of the DIRMAINT machine's minidisks be sent to your virtual card reader.

Operands

filename

specifies the file name of the desired file.

filetype

specifies the file type of the desired file.

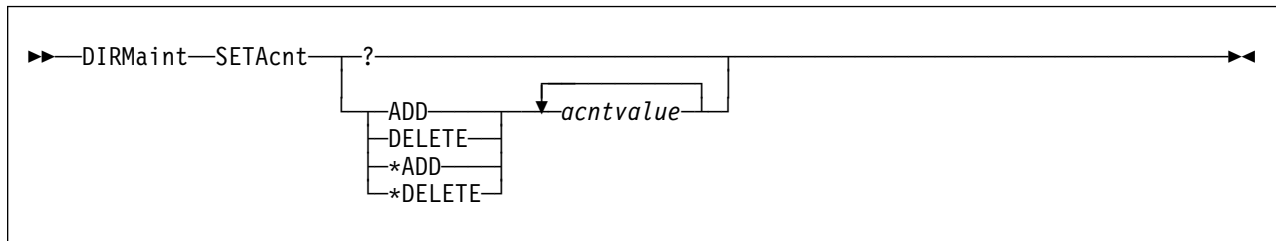
filemode

specifies the minidisk on which the file resides. If *filemode* is omitted, the default value is *, or the first file mode in the DIRMAINT machine's search order where the file occurs.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The requested file will be sent to your virtual card reader. It will be transmitted by the CMS SENDFILE command.

SETACNT



VM Systems: All

Purpose

The SETACNT operand of the DIRMAINT command can be used to manipulate secondary account numbers on an ACCOUNT statement or the *AC= statement.

Operands

? requests that the current secondary account settings be returned in a message.

ADD

DELETE

requests that the account numbers that follow be added to, or deleted from, the secondary list of account numbers on the ACCOUNT statement.

*ADD

*DELETE

requests that the account numbers that follow be added to, or deleted from the list of account numbers on the *AC= statement.

acctvalue

the account number (or numbers) to be added or deleted.

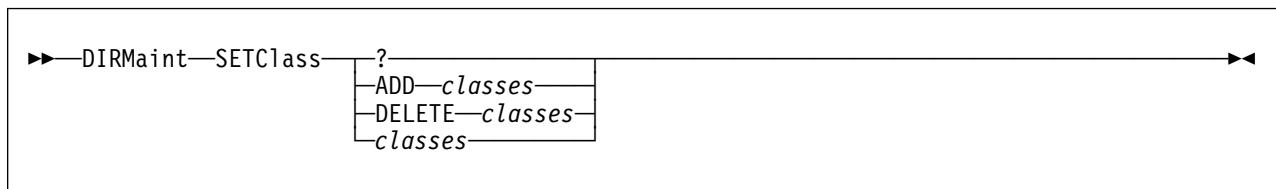
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the ACCOUNT directory statement, see *z/VM: Planning and Administration*.
3. When attempting to add account numbers to the ACCOUNT statement or the *AC= statement, the request is rejected if the new account numbers already exist on the existing statement.
4. When attempting to add account numbers to the ACCOUNT statement, the request is rejected if the total number of account numbers would exceed the maximum allowed on the ACCOUNT statement.
5. Since the *AC= statement is a DirMaint construct, it is not bound by the maximums enforced by the CP ACCOUNT directory statement.

SETACNT

6. When attempting to delete account numbers from the ACCOUNT statement or the *AC=, the request is rejected if the account numbers being deleted do not exist on the current statement.

SETCLASS



VM Systems: All

Purpose

The SETCLASS operand of the DIRMAINT command enables you add, delete, replace, or query the current CLASS value for a user directory entry.

Operands

? allows you to query the current class values.

ADD *classes*

DELETE *classes*

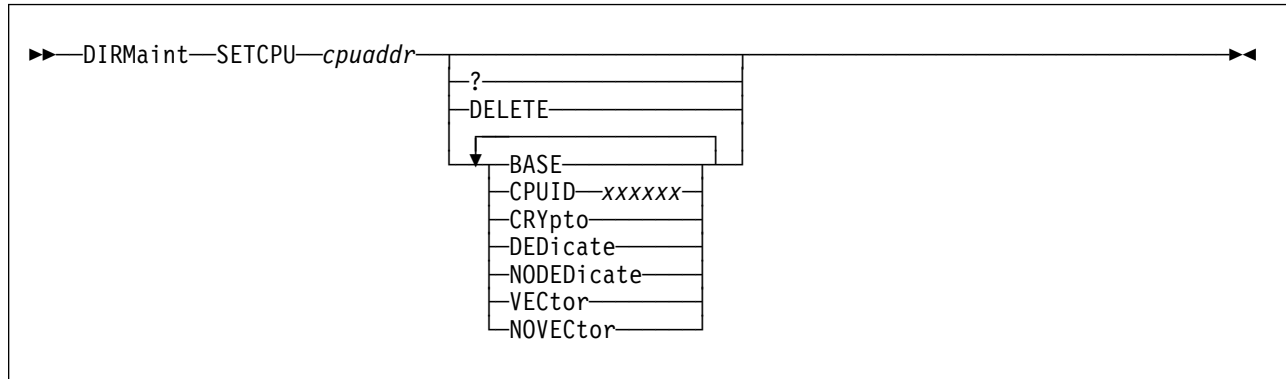
specifies that the listed class values are to be added to, or deleted from, the existing class values.

classes

specified without the ADD or DELETE keyword, the *classes* replace the current class information in the user directory entry.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the CLASS directory statement, see *z/VM: Planning and Administration*.
3. If the keyword ADD and DELETE are not specified (that is, only a list of classes is provided) a REPLACE is assumed.
4. An attempt to add a CLASS value that already exists on the user statement will be rejected.
5. An attempt to delete a CLASS value from the user statement that does not exist will be rejected.

SETCPU


VM Systems: VM/ESA (ESA Feature)

Purpose

The SETCPU operand of the DIRMAINT command allows a user to add, change, delete, or query the central processing unit (CPU) statement in a directory entry.

Operands

cpuaddr

specifies the address of the virtual processor being defined, changed, or deleted. The processor address can be any hexadecimal value from 0 to 3F.

? issues a query against the specified CPU statement.

DELETE

deletes the specified CPU statement from the directory entry.

BASE

specifies that the CPU identified on this statement is defined as the base CPU.

CPUID xxxxxx

specifies the processor ID number that is stored in bits 8 to 31 of the CPU ID and is returned in response to the STIDP instruction.

CRYpto

specifies that a virtual integrated cryptographic facility is defined automatically for the virtual CPU at logon time.

DEDicate

NODEDicate

specifies whether this virtual processor is to be dedicated at logon time to a real processor at logon time.

VEctor

NOVEctor

specifies whether a Vector Facility is assigned for the virtual CPU at logon.

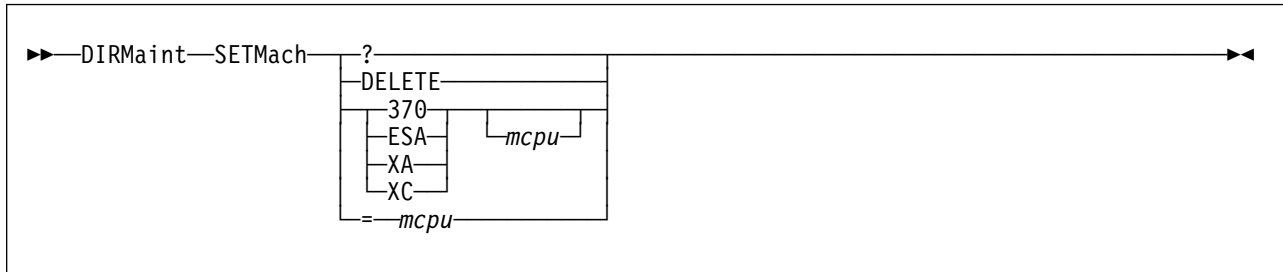
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the CPU directory statement, see *z/VM: Planning and Administration*.
3. Successful execution of the DIRMAINT SETCPU operand depends on the MACHINE directory statement setting defining the maximum number of virtual processors allowed to be defined (indicated by the *mcpu* value). The following rules apply:
 - Modification of an existing CPU statement is always valid. However, only one CPU statement can be given the BASE attribute.
 - Deletion of an existing CPU statement is always valid.
 - The CPU operand is equivalent to the SETCPU operand except that CRYPTO, DEDICATE, and NODEDICATE are not valid on the CPU operand.
 - If a MACHINE statement does exist and the *mcpu* field contains a value, this value is the maximum number of allowable CPU statements.

If any of these rules are violated, SETCPU operand processing will terminate and an appropriate error message is issued.

4. A new CPU statement is always added after all existing CPU statements in your directory entry unless your installation has chosen to sort by device addresses. The order of CPU statements determines the assignment of real processors to virtual CPUs.
5. If you define the virtual CPU as CRYPTO and no real Integrated Cryptographic Facility (ICRF) is installed on the real processor at LOGON time, CP issues a message that indicates that no real crypto facility is installed.

SETMACH



VM Systems: VM/ESA (ESA Feature)

Purpose

The SETMACH operand of the DIRMAINT command allows the general user to create, change, or query a MACHINE statement in a directory entry. If a MACHINE statement does not exist in the directory entry, a new statement is created. If a MACHINE statement does exist, the machine type is changed as specified in the SETMACH operand.

Operands

? requests a query be done of the current MACHINE statement.

DELETE

requests that the MACHINE statement be deleted from the directory.

370

defines a 370-mode virtual machine.

ESA

defines an ESA-mode virtual machine.

XA

defines an XA-mode virtual machine.

XC

defines an XC-mode virtual machine.

= allows an existing operand in the specified position to be used in the resulting machine statement.

mcpu

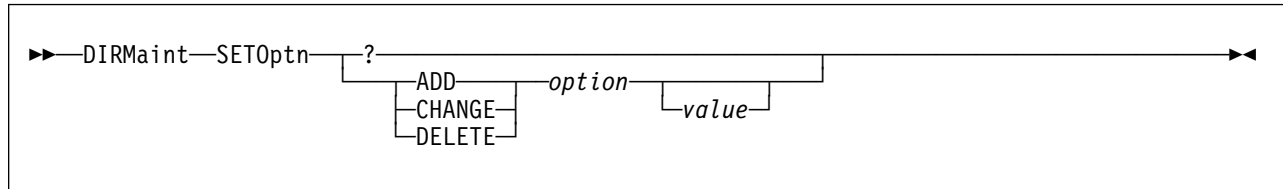
defines the maximum number of CPUs this machine can define.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the MACHINE directory statement, see *z/VM: Planning and Administration*.
3. If the equals (=) operand is used, the *mcpu* operand is required.

4. For more information on the various modes of operation, see the CP SET command in the *z/VM: CP Command and Utility Reference*.
5. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

SETOPTN



VM Systems: All

Purpose

The SETOPTN operand of the DIRMAINT command enables you to request that certain CP options be added to, changed on, or deleted from the OPTION control statement in a directory entry. Once the requested options are in the directory control statement, they will be in effect for each subsequent logon. The SETOPTN operand also allows you to query the OPTION control statement in a directory entry.

Note

Some options can result in a significant increase in overhead. An installation normally exercises restraint in granting use of these options to the general user.

Operands

where:

? requests that the current OPTION statement be returned in a message.

ADD

specifies that the option that follows is to be added to the OPTION control statement in the user's directory.

CHANGE

specifies that the option that follows is to be changed on the OPTION control statement in the user's directory.

DELETE

specifies that the option that follows is to be deleted from the OPTION control statement in the user's directory.

option

specifies one of these options:

ACct

specifies that the virtual machine can issue DIAGNOSE code X'4C' to generate accounting records.

AFfinity

specifies that the virtual machine execution is done on the designated processor identified by the two-digit decimal number between 00 and 63. This operand is not valid on VM/ESA (ESA Feature) systems.

APplmon

specifies that the virtual machine can issue DIAGNOSE code X'DC'.

BMX

specifies that all virtual machine I/O operations are to occur as block multiplexer channel operations. This operand is not valid on VM/ESA (ESA Feature) systems.

COMsrv

authorizes the indicated virtual machine to act as a communication server.

CONceal

places the user in a protected application at logon time.

CFUSER

specifies the user is authorized to connect to a Coupling Facility service machine.

CFVM

specifies the user is a Coupling Facility service machine. CFVM may not be specified with RMCHINFO, V=F, or V=R.

CPuid

provides a processor identification (CPUID) in hexadecimal to be stored in response to the STIDP instruction.

DEDicate

indicates that if this user is logged on as the V=R virtual machine, every virtual processor for this virtual machine is dedicated to a real processor if possible. This operand is not valid on VM/ESA (370 Feature) systems.

DEVInfo

specifies that the virtual machine can issue DIAGNOSE code X'E4', subcodes X'00' and X'01'.

DEVMaint

specifies that the virtual machine can issue DIAGNOSE code X'E4', subcodes X'02' and X'03'.

DIAG200

specifies that the virtual machine can issue DIAGNOSE X'200'. This operand is not valid on VM/ESA (370 Feature) systems.

DIAG88

specifies that the virtual machine can issue DIAGNOSE code X'88'.

DIAG98

specifies that the virtual machine can issue DIAGNOSE code X'98'.

D84NOPAS

specifies that the virtual machine can issue all subfunctions of DIAGNOSE code X'84' except LOGPASS and MDISK, without verification of the existing logon password. This operand is not valid on VM/ESA (370 Feature) systems.

Ecmode

lets this virtual machine run in extended control mode. This operand is not valid on VM/ESA (ESA Feature) systems.

IGNMAXU

indicates that this virtual machine can log on to the system even if the number of users already logged on is equal to or greater than the maximum allowed. This operand is not valid on VM/ESA (370 Feature) systems.

Isam

provides special channel command word translation routines that permit OS/PCP, MFT, and MVT ISAM programs to operate properly in a virtual machine. This operand is not valid on VM/ESA (ESA Feature) systems.

LANG

identifies the language that should be deleted. Used during DELETE.

LANG *langid*

identifies the language that should be set for the virtual machine during logon. Used during ADD and CHANGE.

LINKMSG

indicates a user running disconnected will receive message DMKLN116E when issuing a CP LINK command.

LKFAC

indicates that the specified virtual machine is authorized to use the multipath lock facility RPQ simulation support. This operand is not valid on VM/ESA (370 Feature) systems.

LNKExclu

specifies that the virtual machine is authorized to use the stable and exclusive access modes. This operand is not valid on VM/ESA (370 Feature) systems.

LNKNopas

specifies that the virtual machine is authorized to link to any other virtual machine's DASD without password authorization. This operand is not valid on VM/ESA (370 Feature) systems.

LNKStabl

specifies that the virtual machine is authorized to use the stable access modes. This operand is not valid on VM/ESA (370 Feature) systems.

MAIntccw

authorizes the specified virtual machine to use diagnostic CCWs. This operand is not valid on VM/ESA (370 Feature) systems.

MAXCONN *maxno*

specifies the maximum number of IUCV and APPC/VM connections allowed for this virtual machine.

MAXVMCFI

specifies that the existing MAXVMCFI value should be deleted.

MAXVMCFI *maxno*

specifies the maximum number of VMCF inbound messages, including those initiated by SMSG, plus IDENTIFY final response interrupts that may be queued for processing on this virtual machine.

Mlh

specifies that CP simulate an interrupt for the virtual machine whenever it detects a missing interrupt condition for an I/O operation.

NOChain

Tolerated for migration compatibility from VM/SP HPO. If specified, it will be ignored. This operand is not valid on VM/ESA (370 Feature) systems.

NODEDicate

indicates that if this user is logged on as the V=R virtual machine, no virtual processor for this virtual machine is dedicated to a real processor. This operand is not valid on VM/ESA (370 Feature) systems.

NOMdcfs

specifies that the virtual machine can use minidisk cache at a rate that is not limited by the fair share limit. This operand is not valid on VM/ESA (370 Feature) systems.

NOVf

specifies that the virtual machine may not use the Vector Facility. This operand is not valid on VM/ESA (370 Feature) systems.

Pma

allows an MVS/SP or VSE V=R virtual machine to use preferred machine assist or preferred machine assist with control switch assist. This operand is not valid on VM/ESA (370 Feature) systems.

Quickdsp

causes a virtual machine to be added to the dispatch list immediately when it has work to do.

REaltimer

provides a timer for the virtual machine that is updated during virtual processor runtime and during virtual wait time. This operand is not valid on VM/ESA (ESA Feature) systems.

RMchinfo

specifies that the virtual machine is authorized to access real-machine configuration information. This operand is not valid VM/ESA (370 Feature) systems; and may not be specified with CFVM.

SEtorig

specifies that the virtual machine can issue DIAGNOSE X'F8', subfunction X'00'. This operand is not valid on VM/ESA (370 Feature) systems.

STFirst

specifies that the virtual machine is authorized to use the SET STBYPASS command when virtual machine assist is active on the system for a V=V user. This operand is not valid on VM/ESA (370 Feature) systems.

STGexempt

specifies that the virtual machine is exempt from CP free storage limit detection. This operand is not valid on VM/ESA (370 Feature) systems.

SVCAccl

specifies that the virtual machine can use the Supervisor Call Instruction Acceleration (SVCAccl) when IX/370 is running under CP. This operand is not valid on VM/ESA (370 Feature) systems.

SVCOff

specifies that CP, instead of the virtual machine assist feature or the VM/370 Extended Control Program Support, handles all SVC interrupts for the virtual machine.

SVC76VM

specifies that errors are not recorded by CP. This operand is not valid on VM/ESA (370 Feature) systems.

SVMstat

specifies that the virtual machine is a service virtual machine. This operand is not valid on VM/ESA (ESA Feature) systems.

Todenable

specifies that the user may change the virtual machine's time-of-day clock with the SCK instruction. This operand is not valid on VM/ESA (370 Feature) systems.

VCunoshr

specifies that all devices connected to the virtual machine are to be supported using NONSHARED protocol for virtual I/O operations. This operand is not valid on VM/ESA (ESA Feature) systems.

VIRT=FIXED**V=F**

specifies that this guest is a V=F guest, if possible. This operand is not valid on VM/ESA (370 Feature) systems; and may not be special with CFVM.

VIRT=REAL**V=R**

specifies that this guest is a V=R guest, if possible; and may not be specified with CFVM.

VMsave

specifies that the virtual machine contents are to be saved if CP370 is terminated or if CP370 terminates the indicated virtual machine. This operand is not valid on VM/ESA (ESA Feature) systems.

XMEM

enables MVS cross memory services assist for the MVS/System Product Virtual Machine. This operand is not valid on VM/ESA (370 Feature) systems.

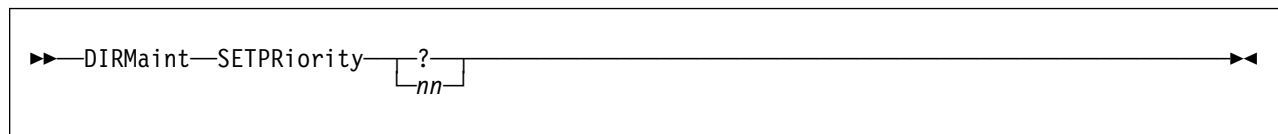
370e

specifies that the MVS/System Extensions support be enabled for the indicated virtual machine. This operand is not valid on VM/ESA (ESA Feature) systems.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOSys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The DirMaint Release 4 SETOPTN command has been renamed to the USEROPTN command for DirMaint Release 5. For more information, see “USEROPTN” on page 264. The SETOPTN command in level 140A performs the same function as in DirMaint Release 4. For level 150A, the SETOPTN command allows the privileged user to add or change the values in the OPTION directory statement.
3. Not every option is valid for every release. For more information on the OPTION directory statement, see *z/VM: Planning and Administration*.

SETPRIORITY



VM Systems: All

Purpose

The SETPRIORITY operand of the DIRMAINT command is used to query or change the CP dispatching priority in a user's directory entry.

Operands

where:

? requests that the current priority be displayed.

nn is a two-digit decimal number that is to be the new CP dispatching priority. The highest priority that can be given a user is 01, the lowest (slowest) is 99. The normal default for the scheduler is 64.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. PRIOSSET and PRIORITY are synonymous with SETPRIORITY. SETPRIORITY is the preferred operand. The others are retained for compatibility with previous releases.
3. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

SETPW

```

▶▶—DIRMaint—SETpw—newpw—┬—VPW—verifypw—┬—nnn—DAYS—┬▶▶

```

VM Systems: All

Purpose

The SETPW operand of the DIRMAINT command is used to change the CP logon password for a user.

Operands

newpw

is the new CP logon password to be in effect for a user.

VPW *verifypw*

specifies the same password again for typographical verification.

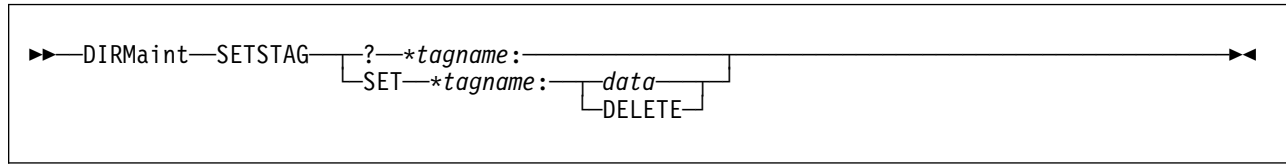
nnn **DAYS**

specifies the number of days till the password expires. *nnn* can be any decimal number from 0 to 366. If not specified, the CONFIG* DATADVH file is consulted and the value specified on the PW_INTERVAL_FOR_SET= entry is used.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Your installation may have rules about password content, minimum length, required frequency of change, or reuse of old passwords. These can be enforced by the PASSWORD_SYNTAX_CHECKING_EXIT exit routine. For more information, see *Directory Maintenance VM/ESA: Tailoring and Administration Guide*. for information on the PASSWORD_SYNTAX_CHECKING_EXIT.
3. A logon password of NOLOG can be set using the SETPW command to prevent a user ID from logging on.
4. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84') if it is enabled.

SETSTAG



VM Systems: All

Purpose

The SETSTAG operand of the DIRMAINT command is used to assign a value to a locally defined tag within a user directory. If the value is already set it may also be queried.

Operands

where:

? requests that the current setting of the specified tag be returned in a message.

***tagname:**
is a locally defined tagname.

SET
requests that the specified tag be changed or added.

data
specifies the data to be assigned to the local tag.

DELETE
indicates that the tag should be deleted from the directory.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. If *data* is provided, its case is respected when placing into the directory.
3. Due to the way DirMaint processes local tags, extra spaces between words are reduced to a single space in local tag data.
4. Data that exceeds a single directory record is split and represented on two lines in the directory entry.
5. The widespread use of local tags has the ability of significantly expanding the size of the source directory. This should be kept in mind when defining local tags and authorizing users to use them.
6. A length limitation is enforced on the data being assigned to the user defined tag. A limit of 60 characters must not be exceeded when the tag name (which is of variable length) is combined with the longest blank delimited token by a single blank. For example, if the stag was:

*MYSTAG:

And the incoming data is:

```
This is my  
Data 123456789!123456789!123456789!123456789!123456789!123456789!
```

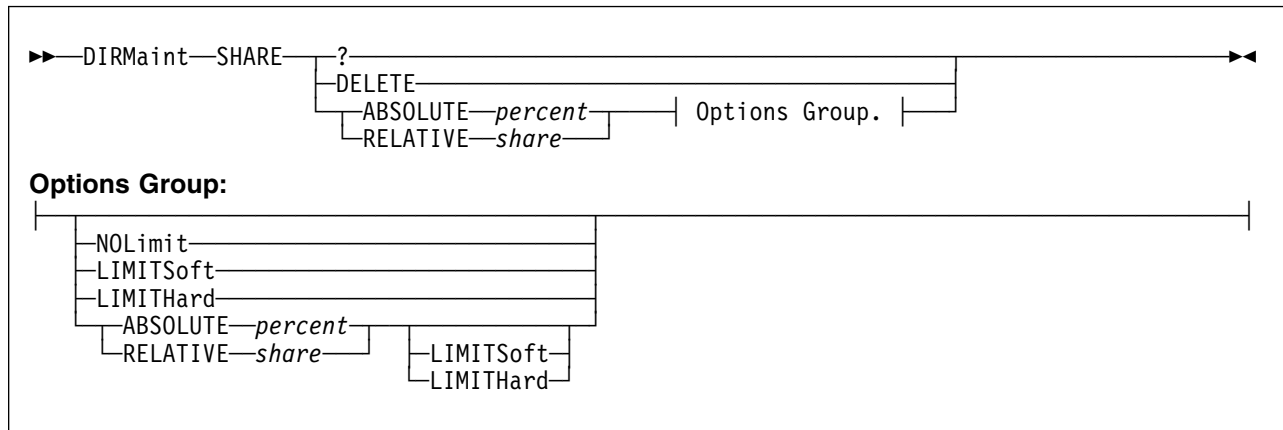
The combination of the tag name and the longest token

```
*MYSTAG: 123456789!123456789!123456789!123456789!123456789!123456789!
```

Would exceed 60 characters and the request would be rejected. If the new data was:

```
This is my  
Data 123456789 123456789 123456789 123456789 123456789 123456789
```

The request would be accepted since no single token would cause the length to exceed 60 characters when combined with the tag name.

SHARE


VM Systems: VM/ESA (ESA Feature)

Purpose

The SHARE operand of the DIRMAINT command can be used to manipulate the CP SHARE directory statement. This statement specifies a virtual machine's share of CPU power.

Operands

? requests that the current share settings be returned in a message.

DELETE

requests that the current share statement be deleted.

ABSOLUTE

specifies a minimum or maximum absolute share of all active processors in a system.

percent

a real number with at most one decimal place, within the range 1 to 100 followed by a percent sign. For example 22.9%.

Note: The decimal place following the value may not be valid on some releases. Consult your system documentation to obtain specific information for your release.

RELATIVE

specifies a minimum or maximum relative share of all active processors in a system.

share

an integer number within the range from 1 to 10000.

NOLimit

specifies that a user's share of processing resource is not limited.

LIMITSoft

specifies that the share of processing resource is limited but at times these users will receive more than their limit.

LIMITHard

specifies that the share of processing resource is limited. These users will not receive more than their limit.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the SHARE directory statement, see *z/VM: Planning and Administration*.
3. When both the minimum value and maximum value for ABSOLUTE or RELATIVE are specified in the same invocation, the minimum value must not exceed the maximum value.

SHUTDOWN

SHUTDOWN

▶▶—DIRMaint—SHUTDOWN—◀◀

VM Systems: All

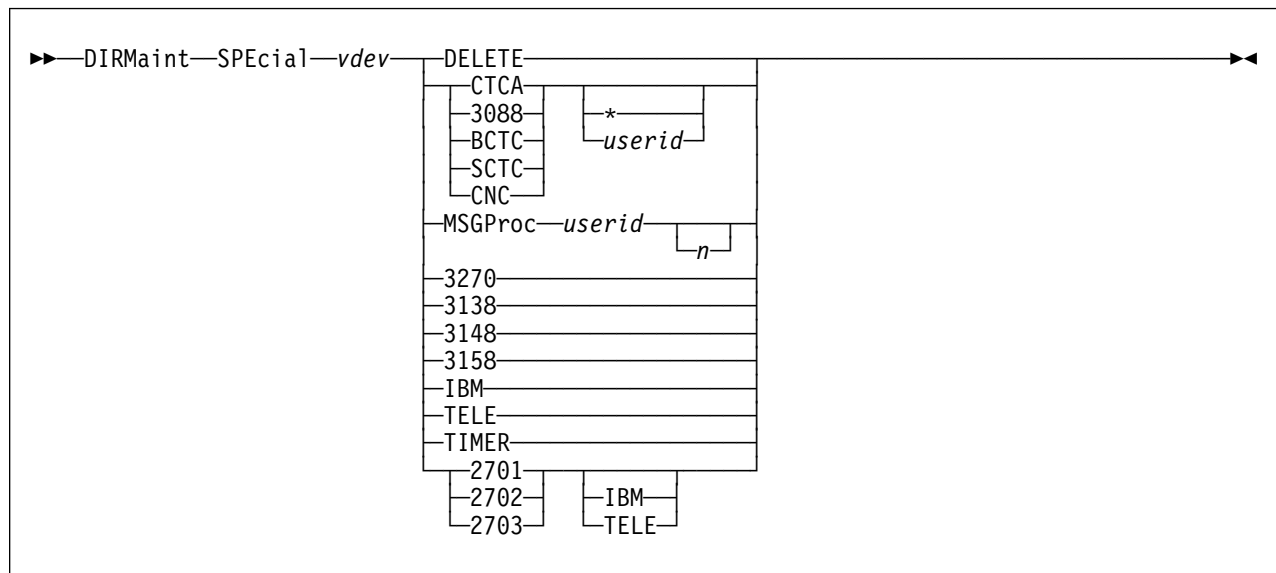
Purpose

The SHUTDOWN operand of the DIRMAINT command is used to shutdown the DIRMAINT virtual machine in an orderly manner.

Usage Notes

1. You may use one or more of the prefix keywords (IMMed, REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. This operand should be invoked a few minutes before a complete shutdown of the system. This gives the DIRMAINT virtual machine time to shutdown in an orderly manner and to log itself off.
3. Any operand currently running will complete normally, and then the DIRMAINT virtual machine will log off if the DIRMAINT machine is running disconnected when the command is received. If the DIRMAINT machine is running connected, the machine will restart and leave the console at the ready prompt. Any requests still queued on disk will remain queued and will be executed after the next startup.

SPECIAL



VM Systems: All

Purpose

The SPECIAL operand of the DIRMAINT command allows the user to add or delete a SPECIAL directory statement to a user directory entry.

Operands

vdev

is the virtual device number for the special device.

DELETE

requests that the SPECIAL statement be deleted.

CTCA

3088

specifies either CTCA or 3088 for a virtual 3088 Multisystem channel Communication Unit logical channel adapter.

BCTC

SCTC

CNC

specifies either a BCTC, SCTC, or CNC virtual communications adapter.

- * tells CP that coupling is to be allowed only from another virtual communications adapter owned by the same virtual machine that owns the virtual communications adapter defined by this SPECIAL statement.

userid

identifies the user ID of a virtual machine allowed to connect to this virtual communications adapter using the CP COUPLE command.

MSGProc *userid n*

defines a virtual message processor and *n* associated message devices in the virtual I/O configuration. It creates a message facility environment for

SPECIAL

the user and establishes a connection to the specified Coupling Facility (CF) Service Machine supplying the message path function. The virtual message processor will only be defined if these conditions exist:

- OPTION CFUSER or OPTION CFVM must be specified in the directory entry for this usage. The *vdev* specified must be the first of four available consecutive device numbers in the user's virtual configuration.
- The CF Service Machine user ID specified must be running prior to this virtual machine.
- The *n* specifies the number of message devices that will be created. The device numbers will be *vdev*, *vdev+1*, and so forth up to to *vdev+n-1*.

If this SPECIAL MSGPROC statement appears in a:

- CFUSER directory entry, this must be in the range of 4 to 16, and must be even. This defaults to 4. If the CFUSER is a 390-mode virtual machine, the *n* you specify will be ignored and 4 will be used instead.
- CFVM directory entry, this must be in the range of 2 to 8. This defaults to 2. If the CFVM is a 390-mode virtual machine, the *n* you specify will be ignored and 2 will be used instead.

3270

is the value for the 3270 display device.

3138

3148

3158

is the virtual console device.

IBM

TELE

is the virtual device type of the line you are defining.

TIMER

is the pseudo timer device.

2701

2702

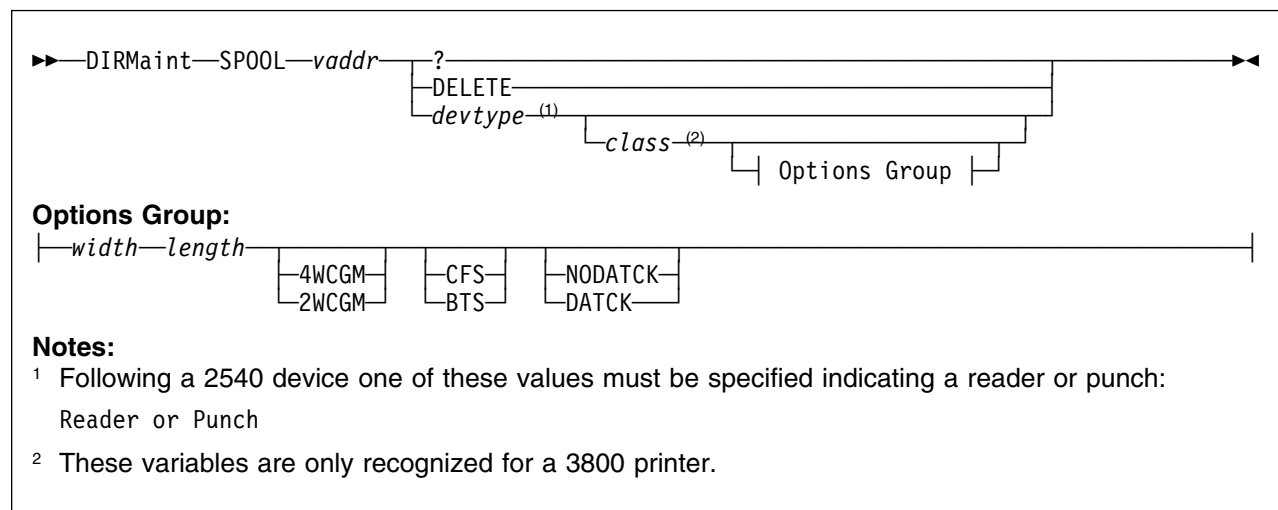
2703

is the value for a communication line.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. All options may not be valid for all releases of VM/ESA. For more information on the SPECIAL directory statement, see *z/VM: Planning and Administration*.

SPOOL



VM Systems: All

Purpose

The SPOOL operand of the DIRMAINT command allows you to change or define virtual unit record device statements.

Operands

vaddr

is the virtual device number for the spooling device.

devtype

is the device type. Valid device types are:

| | | | |
|------|------|---------|--------|
| 1403 | 3289 | 3800 | PCH |
| 2501 | 3505 | 3800-1 | PUnch |
| 3203 | 3525 | 3800-3 | RDR |
| 3211 | 4245 | Printer | Reader |
| 3262 | 4248 | PRT | VAFP |

class

a 1 digit alphanumeric character from A to Z, from 0 to 9, or an asterisk (*) that indicates the spooling class. An asterisk (*) is only valid for a reader device.

width length

specifies the physical characteristics of the paper to be loaded into the 3800 printer. The variable *width* is the hexadecimal form width code of the paper and *length* indicates the decimal length of the paper.

2WCGM

4WCGM

specifies the number of writable character generation modules (WCGM) for the virtual 3800 printer. A WCGM is a 64-position portion of the 3800's character generation storage that holds scan elements of one character set.

SPOOL

A 3800-1 can have either two or four WCGMs. A 3800-3 has four WCGMs.

BTS

CFS

specifies the type of stacker for the virtual 3800 printer. You may specify either CFS (continuous forms stacker) or BTS (burster trimmer stacker).

DATCK

NODATCK

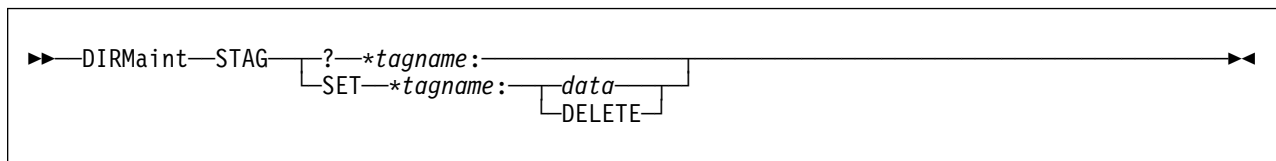
specifies whether CP processes certain virtual 3800 data checks for the virtual machine.

Note: Specifying DATCK severely increases overhead associated with simulation of Write and Skip CCWs to the virtual 3800. Specify DATCK only when absolutely necessary.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the SPOOL directory statement, see *z/VM: Planning and Administration*.

STAG



VM Systems: All

Purpose

The STAG operand of the DIRMAINT command is used to assign a value to a locally defined tag within a user directory. If the value is already set it may also be queried.

Operands

where:

? requests that the current setting of the specified tag be returned in a message.

***tagname:**
is a locally defined tagname.

SET
requests that the specified tag be changed or added.

data
specifies the data to be assigned to the local tag.

DELETE
indicates that the tag should be deleted from the directory.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. If *data* is provided, its case is respected when placing into the directory.
3. Due to the way DirMaint processes local tags, extra spaces between words are reduced to a single space in local tag data.
4. Data that exceeds a single directory record is split and represented on two lines in the directory entry.
5. The widespread use of local tags has the ability of significantly expanding the size of the source directory. This should be kept in mind when defining local tags and authorizing users to use them.
6. A length limitation is enforced on the data being assigned to the user defined tag. A limit of 60 characters must not be exceeded when the tag name (which is of variable length) is combined with the longest blank delimited token by a single blank. For example, if the stag was:

*MYSTAG:

And the incoming data is:

```
This is my  
Data 123456789!123456789!123456789!123456789!123456789!123456789!
```

The combination of the tag name and the longest token

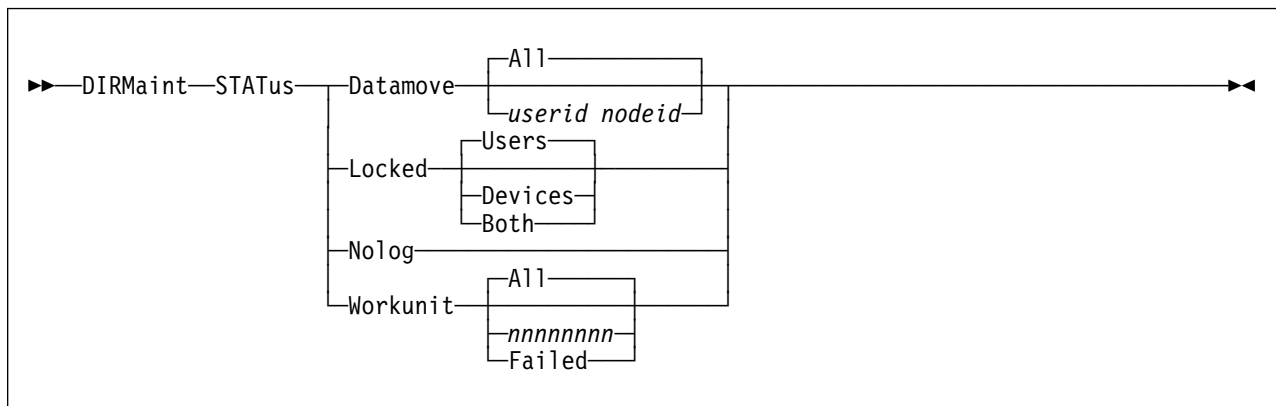
```
*MYSTAG: 123456789!123456789!123456789!123456789!123456789!123456789!
```

Would exceed 60 characters and the request would be rejected. If the new data was:

```
This is my  
Data 123456789 123456789 123456789 123456789 123456789 123456789
```

The request would be accepted since no single token would cause the length to exceed 60 characters when combined with the tag name.

STATUS



VM Systems: All

Purpose

The STATUS operand of the DIRMAINT command is used to request various status information about the DIRMAINT or DATAMOVE virtual machines.

Operands

Datamove

specifies a request for the current status of the DATAMOVE virtual machine(s).

Locked

specifies a request for a list of those user IDs whose directory entry is locked (to prevent simultaneous updates).

Note: Users are locked by a specific invocation of the LOCK or GET operands and can only be unlocked by the UNLOCK or REPLACE operands.

Users

requests that all locked users be reported. This is the default.

Devices

requests that all locked devices be reported.

Both

requests that all locked users and devices be reported.

Nolog

specifies a request for a list of those user IDs whose logon passwords are NOLOG.

userid nodeid

requests that a specific DATAMOVE machine be listed.

Workunit

requests a status a Work Unit Control Files (WUCF) from the DASD control subsystem.

STATUS

All requests that all current active WORKUNITS or DATAMOVE machines be listed. This is the default.

nnnnnnnn

requests the status of a specific WORKUNIT (identified by the passed eight digit identifier) be reported. This value may represent an ACTIVE or FAILED workunit.

Failed

requests that a list of all failed WORKUNIT's be reported.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. A user is locked to prevent concurrent updates to the source while an administrative activity is underway. A user can be locked with an explicit DIRM LOCK command or by a DIRM GET (without the NOLOCK option). The user is unlocked when a DIRM REPLACE command is processed or by an explicit DIRM UNLOCK command.
3. A device is locked to indicate a pending activity on the device. An example of this is a pending AMDISK request for a minidisk address. The device will be unlocked when the AMDISK request completes successfully (or when it is rolled back during a failure). The lock may remain active if an AMDISK fails and DirMaint was unable to roll back the transaction. The device may be unlocked by an explicit DIRM UNLOCK *addr* command.
4. The existence of a failed WORKUNIT indicates that a work unit was created for a task that could not be accomplished. DIRMAINT will take steps to "roll back" any changes made by this work unit under many circumstances. The Work Unit Control File is maintained for detailed study by the system administrator. The contents of the failed workunit can be determined using the WORKUNIT option of this command. The failed workunit is maintained on the primary directory disk (the 01DF by default) on the DIRMAINT server as:

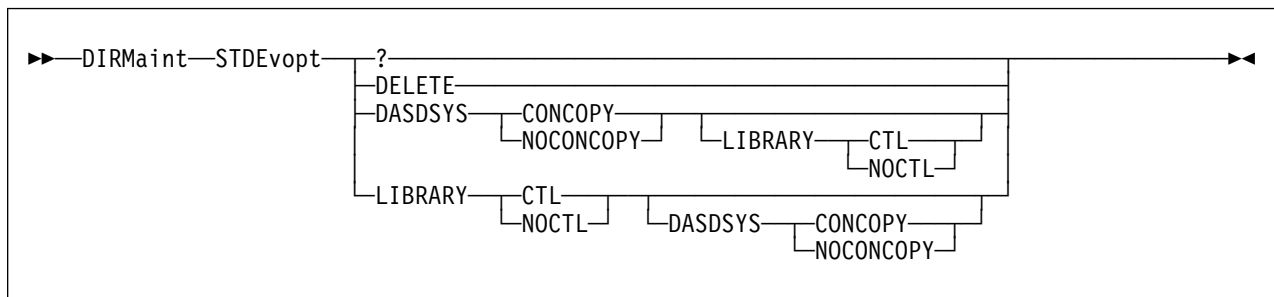
nnnnnnnn WUCFFAIL

Active workunits are maintained on the same disk on the DIRMAINT server as:

nnnnnnnn WORKUNIT

where *nnnnnnnn* is the workunit number in both cases.

STDEVOPT



VM Systems: All

Purpose

The STDEVOPT operand of the DIRMAINT command allows you to specify the optional storage device management functions available to a virtual machine.

Operands

? requests that the current STDEVOPT directory control statement specifications for the virtual machine be returned in a message.

DELETE

deletes any STDEVOPT directory entries for the virtual machine.

DASDSYS CONCOPY

DASDSYS NOCONCOPY

tells CP whether the virtual machine is authorized to control and process Concurrent Copy and Peer To Peer Remote Copy Establish Pair CCW.

LIBRARY CTL

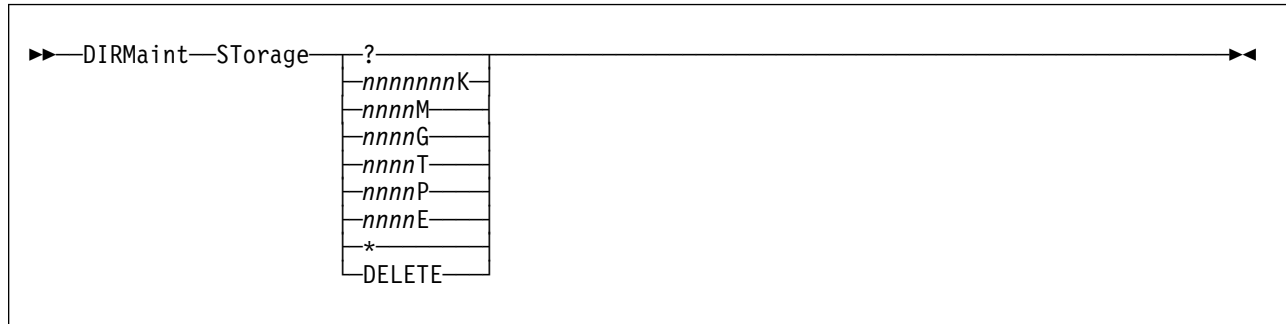
LIBRARY NOCTL

tells CP whether the virtual machine is authorized to control a 3495 Tape Library Dataserver.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the STDEVOPT directory statement, see *z/VM: Planning and Administration*.

STORAGE



VM Systems: All

Purpose

The STORAGE operand of the DIRMAINT command is used to set or display the current virtual machine storage size on the USER statement.

Operands

?

nnnnnnnK

specifies the maximum value allowed in kilobytes.

nnnnM

specifies the maximum value allowed in megabytes.

nnnnG

specifies the maximum value allowed in gigabytes.

nnnnT

specifies the maximum value allowed in terabytes.

nnnnP

specifies the maximum value allowed in petabytes.

nnnnE

specifies the maximum value allowed in exabytes.

***** specifies when used within a:

- USER directory entry, the value specified within the included PROFILE entry will be used if present, or the CP default value of 1M will be used if there is no STORAGE value within the included PROFILE; and
- PROFILE directory entry, the CP default value of 1M will be used unless the including USER entry specifies an overriding STORAGE value.

DELETE

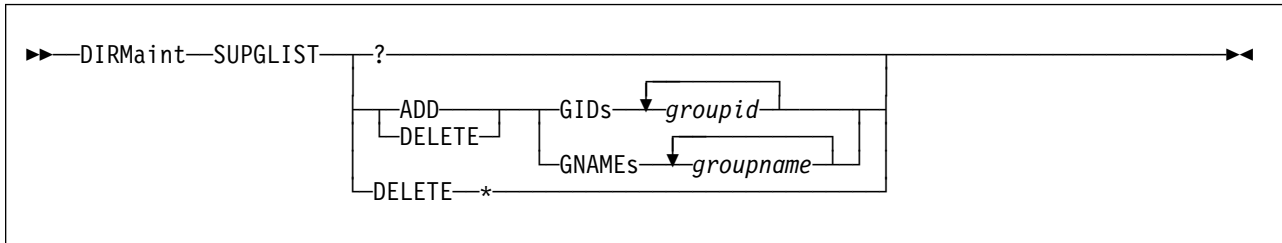
specifies that the STORAGE statement will be deleted, resulting in use of the:

- Value specified within an included PROFILE if deleted from a USER directory entry, or use of the CP default value of 1M if there is no STORAGE value within the included PROFILE

- CP default value of 1M when deleted from a PROFILE entry, unless the including USER entry specifies an overriding STORAGE value.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the STORAGE directory statement, see *z/VM: Planning and Administration*.
3. The suffix K, M, G, T, P or E is required.
4. A query request will report the default values for the default storage size and maximum storage size if they are not specified on the user statement.
5. The value is placed into the directory as provided once it has been validated for correct format. Any required rounding is done by CP.
6. If the directory definition of the user ID being modified does not currently have default or maximum virtual storage sizes defined, a default logon storage size of 256KB is assigned to the user ID. This value becomes the minimum that can be specified for the maximum virtual storage size.
7. The values supplied must be whole numbers. Values such as 4096.8K or 6.5M are not supported.
8. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

SUPGLIST


VM Systems: All

Purpose

The SUPGLIST operand of the DIRMAINT command is used to add, delete, or query Group IDs or Group Names on the DirMaint supplemental group list.

Operands

? requests that the contents of the current supplemental group list be returned in a message.

ADD

DELETE

requests that the specified value be added to, or deleted from the current supplemental group list.

DELETE *

requests that the entire statement be deleted.

GIDS *groupid*

specifies that the following values are to be considered group IDs (GIDs).

GNAMEs *groupname*

specifies that the following values are to be considered group names (GNAMEs).

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. DirMaint normally represents POSIXGLIST's as a single POSIXGLIST statement with all GIDs preceding GNAMEs. Regardless of their position in the directory entry, they are placed on a single record.

In some environments this may have undesirable effects. The user's supplementary GID list consists of up to 32 unique GIDs. The primary GID (from the POSIXINFO statement) is always part of this list. The remainder of the supplementary list is built by consulting the users POSIXGLIST statements, in order, from the directory entry. If the directory entry has more than 32 items listed, the supplementary list may be built incorrectly due to the method used by DirMaint to represent the POSIXGLIST internally.

By adding a GID or GNAME to the supplemental Glist, DirMaint will generate a POSIXGLIST statement with these GIDs and GNAMEs first, and any remaining GIDs and GNAMEs will follow on a subsequent POSIXGLIST statement. Using this method a user can choose which groups will be specified in a position to be used on their supplemental list.

3. When adding a GID or GNAME, the request is rejected if the value already exists on the list.
4. When deleting a GID or GNAME, the request is rejected if the value is not found on the current list.
5. Adding a value to the supplemental Glist that is not on the actual POSIXGLIST does not result in an actual POSIXGLIST entry. Remember, the supplemental Glist is a DirMaint construct (its internal representation is a comment within the directory) that controls how the actual POSIXGLIST statement is generated in the resultant directory. If a GID or GNAME is present on the supplementary Glist and is also on the actual POSIXGLIST, it is placed in the first generated POSIXGLIST. If an entry is placed on the supplemental Glist that does not exist on the actual POSIXGLIST, no actual POSIXGLIST is generated. For example, if the internal representation of the directory is:

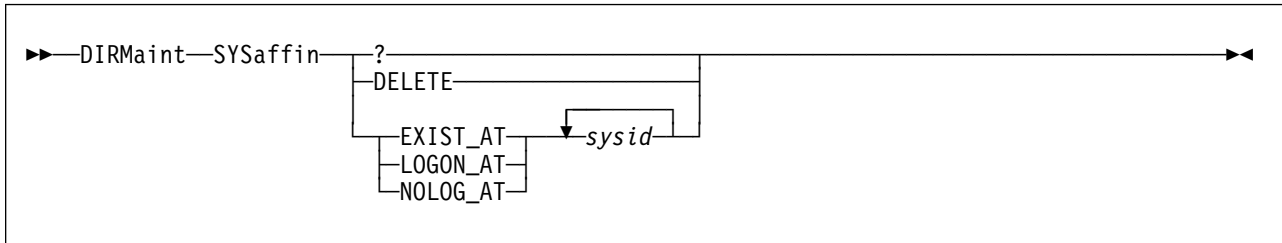
```
-
-
*DVGHLIST= GID 1 GNAME MyG2 MyG3 Unk1 Unk2
POSIXGLIST GID 1 2 3 GNAME MyG1 MyG2 MyG3 MyG4
-
-
```

The resultant directory will look as follows:

```
-
-
*DVGHLIST= GID 1 GNAME MyG2 MyG3 Unk1 Unk2
POSIXGLIST GID 1 GNAME MyG2 MyG3
POSIXGLIST GID 2 3 GNAME MyG1 MyG4
-
-
```

Note how all items that were on the DirMaint supplemental Glist (*DVGHLIST=) and also on the actual Glist ended up on the first generated Glist, all remaining values were represented on the second Glist. The values from the DirMaint supplemental Glist without a corresponding Glist entry remain as comments.

SYSAFFIN



VM Systems: VM/ESA (ESA Feature)

Purpose

The SYSAFFIN operand of the DIRMAINT command is used to support the prefix form of System Affinity.

Operands

? requests that the current values associated with the prefix form of System Affinity be returned in a message.

DELETE

requests that the current prefix System Affinity statement be deleted.

EXIST_AT

specifies that the following user definition is to be compiled into the object directory only when running one of the listed systems.

LOGON_AT

specifies that the following user definition is to be compiled into the object directory but the associated virtual machine will only operate on the specified systems. It will be NOLOGGED on all other systems.

NOLOG_AT

specifies that the following user definition is to be compiled into the object directory but the associated virtual machine will not operate on the specified systems. However, the machine and all its resources are to exist as directory entries in the object directories of all machines.

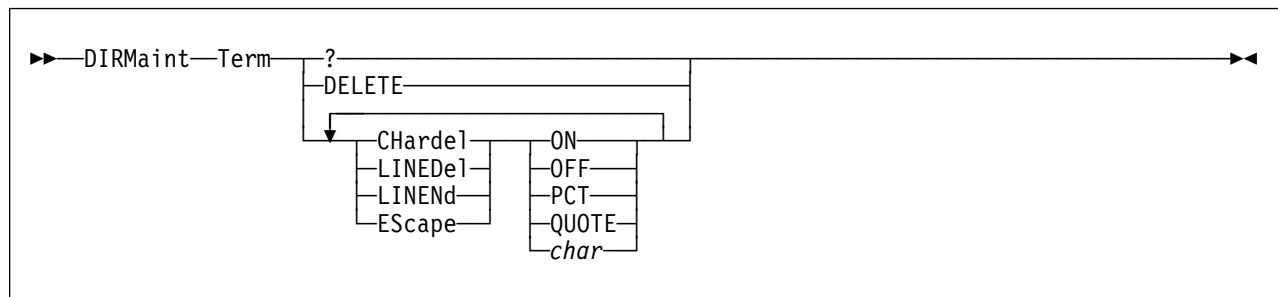
sysid

a system ID associated with the prefix format of System Affinity.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the SYSAFFIN directory statement, see *z/VM: Planning and Administration*.
3. Use the DirMaint ATnode prefix operand to specify the internal form of System Affinity.
4. This operand is only valid on USER directory entries.

5. An attempt to use this operand on a directory entry that has the internal format of System Affinity is tolerated. However, the operation is rejected if a conflicting definition results.

TERM


VM Systems: All

Purpose

The TERM operand of the DIRMAINT command allows you to establish customized logical line editing symbols, or modify an existing set of symbols on the USER control statement. These symbols are then in effect for all terminal sessions.

Operands

CHardel

Indicates that one of the following options applies to the character delete symbol:

- ON** specifies that the default character delete symbol is @.
- OFF** specifies that no character delete symbol is to be active.
- PCT** Indicates that editing character is to be a percent sign (%).
- QUOTE** Indicates that editing character is to be a single quote (').
- char* indicates that the value specified for *char* is to become the character delete symbol.

LINEdel

Indicates that one of the following options applies to the logical line delete symbol:

- ON** specifies that the default line delete symbol is ␣.
- OFF** specifies that no logical line delete symbol is to be active.
- PCT** Indicates that editing character is to be a percent sign (%).
- QUOTE** Indicates that editing character is to be a single quote (').
- char* indicates that the value specified for *char* is to become the logical line delete symbol.

LINENd

Indicates that one of the following options applies to the logical line end symbol:

- ON** specifies that the default logical line end symbol is #.
- OFF** specifies that no logical line end symbol is to be active.

PCT Indicates that editing character is to be a percent sign (%).

QUOTE Indicates that editing character is to be a single quote (').

char indicates that the value specified for *char* is to become the logical line end symbol.

EScape

Indicates that one of the following options applies to the logical escape symbol:

ON specifies that the default logical escape symbol is ".

OFF specifies that no logical escape symbol is to be active.

PCT Indicates that editing character is to be a percent sign (%).

QUOTE Indicates that editing character is to be a single quote (').

char indicates that the value specified for *char* is to become the logical escape symbol.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The CP TERMINAL command allows you to establish logical line editing symbols that will then be in effect for the remainder of your terminal session. These symbols can be used for character deletion, line deletion, line end, or escape character purposes.

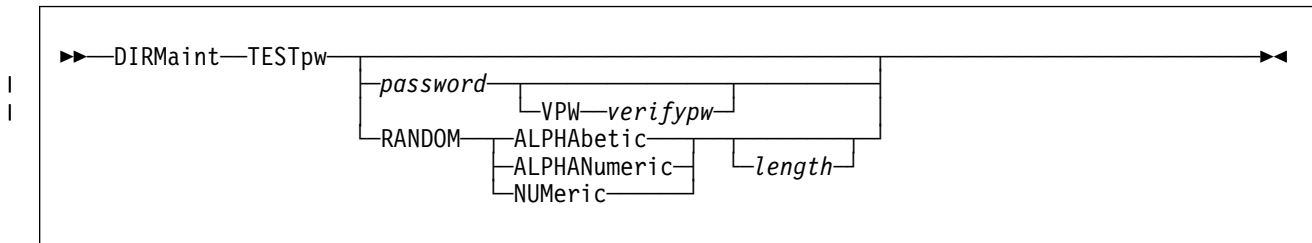
For more information on editing symbols, see the CP TERMINAL command in the see *z/VM: CP Command and Utility Reference*.

3. You can specify more than one option in a single DIRMAINT command. For example, the following command would turn on the standard character delete symbol @, set the escape character to !, set the line end character to %, and the line delete character to '.

```
DIRM TERM CH ON ES ! LINEND PCT LINEDEL QUOTE
```

Note that PCT is used for the percent symbol (%) and QUOTE is used for the single quotation mark (').

4. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you receive a message informing you that the source and online directories have been updated.

TESTPW


VM Systems: All

Purpose

The TESTPW operand of the DIRMAINT command allows users to check a proposed new CP logon password for acceptability with installation policies. Users are prompted to enter their new password if it was not specified with the TESTPW operand. The password is masked out as blanks on the console as they are typed.

The new password is presented to any locally-defined user exit procedure. If the password is not acceptable, the user exit should inform the user and explain why the password was rejected. The TESTPW operand is for screening only, it will not update the password.

Operands

password

specifies the value to be used as a password.

VPW verifypw

specifies the same password again for typographical verification.

RANDOM

requests that a random password be generated by calling the PASSWORD_RANDOM_GENERATOR_USER_EXIT.

ALPHAbetic

requests that the random generator exit be provided with the keyword ALPHABETIC in the requested algorithm field.

ALPHANumeric

requests that the random generator exit be provided with the keyword ALPHANUMERIC in the requested algorithm field.

NUMeric

requests that the random generator exit be provided with the keyword NUMERIC in the requested algorithm field.

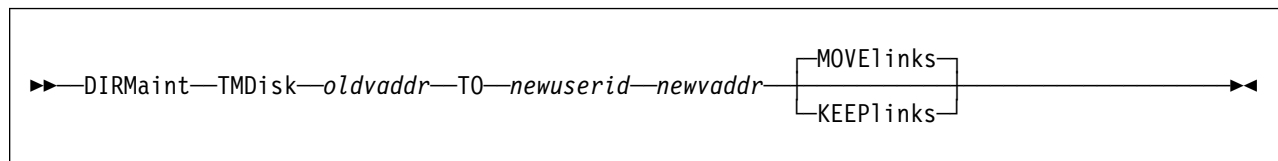
length

requests that the random generator exit be provided with the value *length* as an optional parameter.

Usage Notes

1. You may use one or more of the prefix keywords (IMMed, REQuest, TOsys, ASuser, BYuser, FORuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. As part of the verification of passwords, the PASSWORD_SYNTAX_CHECKING_USER_EXIT routine is called.

For more information, see *Directory Maintenance VM/ESA: Tailoring and Administration Guide*.
3. Passwords generated by a call to the PASSWORD_RANDOM_GENERATOR_USER_EXIT are still passed through the PASSWORD_SYNTAX_CHECKING_USER_EXIT for verification.
4. Your installation may maintain a list of “restricted passwords,” that is, specific passwords whose use are restricted. The transaction will be rejected with an error message if your new password is on the restricted list.
5. All passwords are checked by the installations PASSWORD_SYNTAX_CHECKING_EXIT routine, even though they have already passed the checking done by the USER exit.

TMDISK


VM Systems: All

Purpose

The TMDISK operand of the DIRMAINT command is used to transfer ownership of a minidisk from one user to another.

Operands

oldvaddr

is the virtual device address of the minidisk in the current user's configuration.

TO

is a required keyword.

newuserid

is the user ID who is to be given the minidisk.

newvaddr

is the virtual device address of the minidisk when placed in *newuserid's* configuration.

MOVE|links

requests that all links to the source user's minidisk be deleted and a link to the new user's minidisk be generated in their place.

KEEP|links

requests that all links be left as they are currently defined.

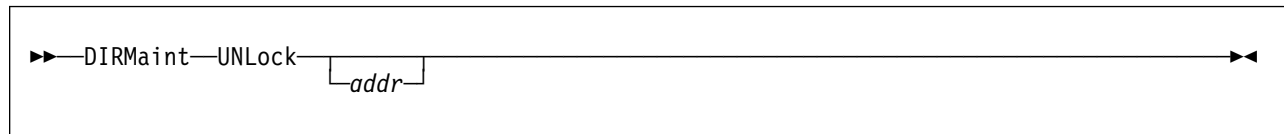
Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. This operand should be used when the minidisk and all files on it are to be transferred to a different user ID.
3. The new owner will inherit all files currently on the minidisk. The minidisk in the new owner's configuration will have the same access mode and access passwords as established by the original owner.
4. If the intent is merely to transfer the space, then the original owner should be told that they should initialize or reformat the disk before the transfer is done. This may merely be a matter of individual privacy, or may involve company policy or security. Individual installations may emphasize this procedure as appropriate.

5. Any links to the transferred minidisk are changed to link to the new owner's minidisk if the MOVELINKS option is used. This is done as low priority work element (batch). If the "linker" issues a LINK DELETE request before the low priority command executes, the low priority command to change the link to the new owner will fail. This "error" condition can be safely ignored.

UNLOCK

UNLOCK



VM Systems: All

Purpose

The UNLOCK operand of the DIRMAINT command is used to remove a lock against a directory entry or remove a lock against a specific minidisk within a directory entry.

Operands

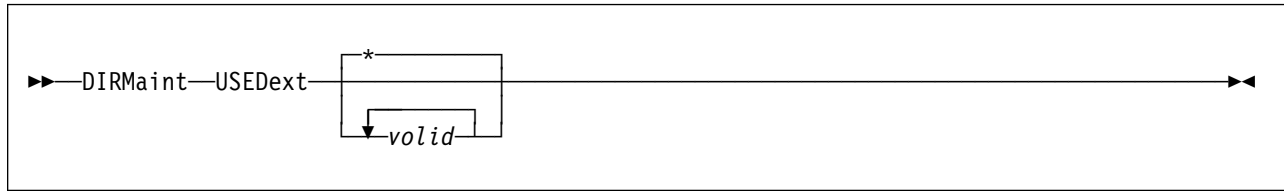
addr

is the virtual address of the minidisk to be unlocked.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. There are two circumstances that lead to the setting of a user ID or profile lock:
 - a. Use of the GET operand (without the NOLOCK option)
 - b. Explicit use of the LOCK function.
3. The UNLOCK operand can be used whenever it is desirable to allow a user to resume making changes directly to their own directory entry.
4. The UNLOCK operand will be invoked automatically whenever the REPLACE operand is used.
5. The *addr* option provides support to unlock a minidisk. Device locks are set by REPLACE, AMDISK, CMDISK, and DMDISK operands.

USEDEXT



VM Systems: All

Purpose

The USEDEXT operand of the DIRMAINT command is used to generate a listing of allocated extents on volumes used to contain minidisks.

Operands

valid

specifies the search domain.

If a 1 to 6 character volume label is specified, the listing will contain only allocated extents on that volume.

If a partially-qualified volume label of less than 6 characters is specified followed by an asterisk (*), the listing will include allocated extents on all volumes beginning with the qualified *valid*. For example, if the total set of volumes used to contain minidisks is:

```
ABC
ABC12
ABC456
ABX445
VOL001
VSZ777
```

then specifying *valid* as ABC* will cause volumes ABC, ABC12 and ABC456 to be scanned.

If *valid* is omitted, or if an * is specified, then all volumes used to contain minidisks will be included in the output.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. Processing consists of making one pass over each volume control file (one per *valid*) and listing all MDISK control statements found that are allocated on volumes within the search domain defined by *valid*. As each disk in the candidate list is added to the report, its starting cylinder or block number is sorted.
3. The name of the file returned to the invoker of this command is dependant on the method used to invoke the command.

valid USEDEXT

if the invocation was made with a specific volume specified.

`vo$ USEDEXT`

if the invocation was made with a partially qualified valid specified.

`ALLVOLS USEDEXT`

if the invocation was made with an `*` as the qualifier.

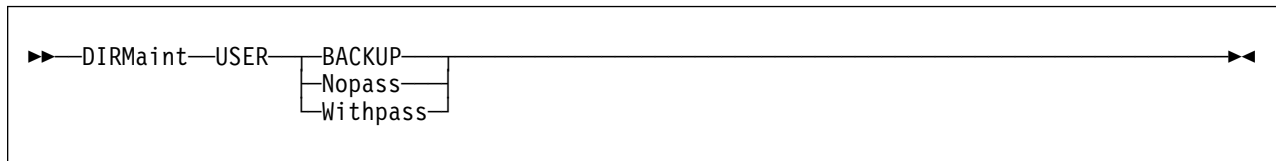
`MULTIVOL USEDEXT`

if the invocation was made with more than one qualifier.

Note: When there are no free extents or used extents meeting the search criteria the corresponding file returned will contain only a header with no data.

4. In the normal case, the starting cylinder or block for one minidisk will be greater than the ending cylinder or block for the preceding minidisk on the volume. If the starting cylinder or block for any minidisk is less than or equal to the ending cylinder or block of the preceding minidisk, an exception is flagged. If the two minidisks have the:
 - Same starting and ending location, virtual disk address, and owner, differing only in the system affinity associated with the two entries (with neither system affinity being an asterisk), then the condition is flagged as shared.
 - Same starting and ending location, address, owner, differing only in the system affinity with one of them being an asterisk or if the system affinity values are also the same, then the condition is flagged as an error.
 - Different starting locations, or different ending locations, or different virtual addresses, or different owners, then the condition is flagged as an overlap. If any exceptions are flagged, a `DIAGNOSE` file will be generated in addition to the requested report file.

USER



VM Systems: All

Purpose

The USER operand of the DIRMAINT command is used to obtain a copy of the source directory in a single file format. It is also used to generate the USER BACKUP file, which is the primary disk backup of the source directory.

Operands

BACKUP

specifies that a file is not returned to the invoker, but a composite backup file named USER BACKUP is generated on the backup disks. The previous USER BACKUP file is retained and renamed to USER BACKOLD. This feature requires that the backup disk be large enough to contain two copies.

Note: The difference between the BACKUP and the USER BACKUP operands is that the USER BACKUP operand causes the creation of the USER BACKUP file only. The BACKUP operand invokes the full backup procedure which also backs up the auxiliary control files.

Nopass

sends an edited copy of the source directory to your virtual card reader named USER NOPASS. All logon passwords are replaced with XXXXXXXX. All minidisk access passwords are replaced (if they exist) as the:

- Read password is replaced with READPASS
- Write password is replaced with WRITPASS
- Multiple write password is replaced with MULTPASS.

Any APPCPASS statements in the source directory will have the password field set to XXXXXXXX.

Withpass

sends an unmodified copy of the source directory to your virtual card reader named USER WITHPASS.

Note: This file contains all passwords for all users at your installation. Take the appropriate security precautions.

Usage Notes

1. You may use one or more of the prefix keywords (IMMed, REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. The NOPASS option produces an edited version of the source directory that can be printed and kept available for reference. While passwords are not revealed, other items of information may be considered sensitive at your installation. Therefore, you should take appropriate security measures with the printed listing.
3. The USER BACKUP command may be used by authorized users or external processes that require read-only access to the current state of the directory source. After the command completes, staff members may use the DIRM SEND command to request that the newly generated USER BACKUP file be sent to their virtual reader, or they can access the DirMaint 1DB and process the file from there.
4. For the NOPASS and WITHPASS options, the file sent to your virtual reader will be in single file format. It should be read on to your disk using the CMS RECEIVE command.
5. When creating a new source directory from either a USER WITHPASS or USER BACKUP, the previous USER DIRECT file must be erased before executing the DVHBEGIN EXEC otherwise the new USER INPUT file will not be used.

USERMAP



►►—DIRMaint—USERMAP—◄◄

VM Systems: All

Purpose

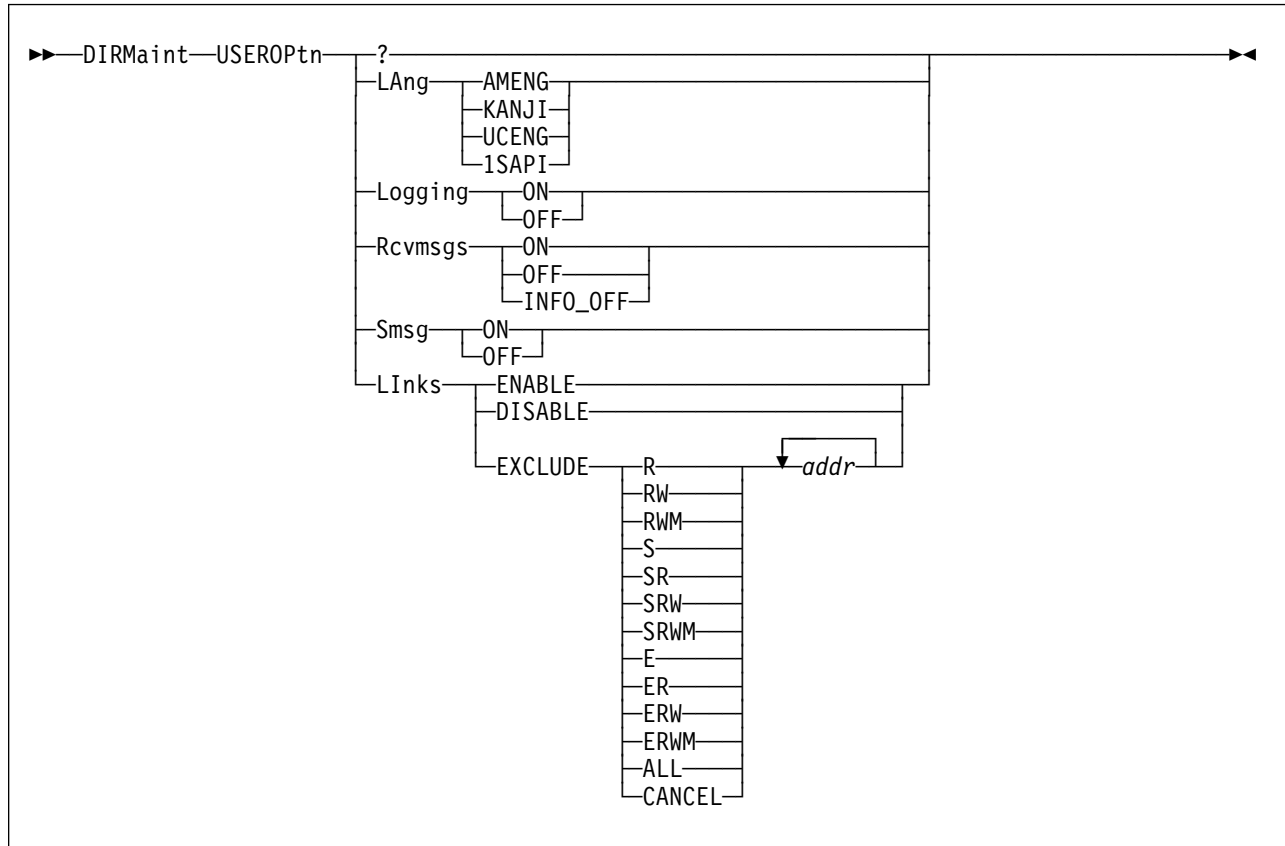
The USERMAP operand of the DIRMAINT command generates a list of profiles and user IDs currently in the source directory. The contents of the file that DirMaint sends is called USER DIRECT. This file is the same as a CP LOAD Control Statement.

For more information on the CP LOAD Control statement, see *z/VM: Planning and Administration*.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. This operand generates a file named USER DIRECT containing all profile names, user IDs, and their starting record numbers within the source directory, and sends it to your virtual card reader. The file can be loaded to your minidisk using the CMS RECEIVE command.

USEROPTN



VM Systems: All

Purpose

The USEROPTN operand of the DIRMAINT command controls the interaction between your virtual machine and the DIRMAINT virtual machine.

Operands

? requests that the current settings be returned in a message.

LAng langid

specifies the default DirMaint language associated with this entry.

Logging

specifies whether the messages sent to your user ID by the DIRMAINT virtual machine are to be logged if they cannot be sent to you. The file name of the log file will be the user ID name, with a file type of the node ID from where the report was created.

Note: The USER LOG identifies the *userid* of the issuer of the command issuer and uses a file type of *nodeid* of that user.

ON specifies messages are to be logged. This is the default however, the default may be different on your system.

OFF

indicates that those messages will be lost.

Rcvmsgs

specifies whether your virtual machine is accepting messages from the DIRMAINT virtual machine.

ON indicates that messages from the DIRMAINT virtual machine be sent to your terminal. This is the default however, the default may be different on your system. If you are not currently receiving messages, but the LOGGING=ON option is in effect, these messages will be logged.

OFF

indicates that messages from the DIRMAINT virtual machine will not be sent to your terminal. OFF automatically invokes the LOGGING=ON option.

Note: The USEROPTN RCVMSGs OFF will not automatically issue the LOGGING=ON option.

INFO_OFF

indicates that informational messages from the DIRMAINT virtual machine will not be sent to your terminal, but will be logged if the LOGGING=ON option is in effect.

Smsg

specifies how messages are to be sent to your virtual machine.

ON indicates sending messages as virtual storage images by means of the CP SMSG facility.

OFF

indicates sending messages in visual form by means of the CP MSG facility. This is the default however, the default may be different on your system.

Links

specifies whether other general users can establish links to your minidisks using the DIRM LINK command.

ENABLE

allows other general users to establish links to minidisks owned by your user ID.

DISABLE

prevents other general users from establishing links to minidisks owned by your user ID. This is the default however, the default may be different on your system.

EXCLUDE mode addr

specifies that any entry for the indicated minidisk be added to, or the existing entry be changed or removed from, the DVHLINK EXCLUDE control file. A single entry for the *addr* minidisk in the DVHLINK EXCLUDE file prevents multiple entries for *mode* links from appearing in the DVHLINKS FILE.

Valid *mode* requests include:

R Read links (R and RR) should be excluded.

RW

Read and Write links (R, RR, W and WR) should be excluded.

RWM

Read, Write and Multi Write links (all except S and E links) should be excluded.

S Stable links (any link using the S suffix) should be excluded.

SR Stable Read links (any read link using the S suffix) should be excluded.

SRW

Stable Read and Write links (any read or write link using the S suffix) should be excluded.

SRWM

Stable Read, Write and Multi Write links (all except Exclusive links) should be excluded.

E Exclusive or Stable links (any link using the S or E suffix) should be excluded.

ER Exclusive or Stable Read links (any read link using the S or E suffix) should be excluded.

ERW

Exclusive or Stable Read and Write links (any read or write link using the S or E suffix) should be excluded.

ERWM

Exclusive or Stable Read, Write and Multi Write links (all links) should be excluded.

ALL

All links should be excluded.

CANCEL

The entry should be removed from the link exclude control file.

An excluded link is not included in DirMaint's link inventory control file. Therefore, links to excluded devices are not:

- Shown in the result of a REVIEW command.
- Changed by a CHNGID or CHVADDR command.
- Moved by a TMDISK command (MOVELINKS option).
- Deleted by a DMDISK command (MOVELINKS option).
- Deleted by a DLINK * command.

To have these commands affect an excluded disk, use the

```
USEROPTN LINKS EXCLUDE CANCEL
```

command to remove the entry from the exclude control file, then run RLDEXTN with the ALL option to rebuild the link inventory control file.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. A listing of the USEROPTN options currently in effect for your user ID is also returned as part of the output from the DIRM REVIEW operand.
3. Options not specified are not changed.
4. Specifying LINKS DISABLE for an individual user will not prevent an administrative staff member from issuing a DIRM LINK command to that user's minidisk. The protection is only against other general users.
5. The DIRMAINT LINKS ENABLE/DISABLE setting has no effect on CP. Issuing the command DIRM USEROPTN LINKS DISABLE does not inhibit the CP LINK command to your minidisk. It is effective only against DirMaint link requests.
6. Use the RLDEXTN command with the ALL option to rebuild the link file after using USEROPTN to alter the links exclude file or wait for nightly automatic backup processing to take place.

VERIFIED

VERIFIED

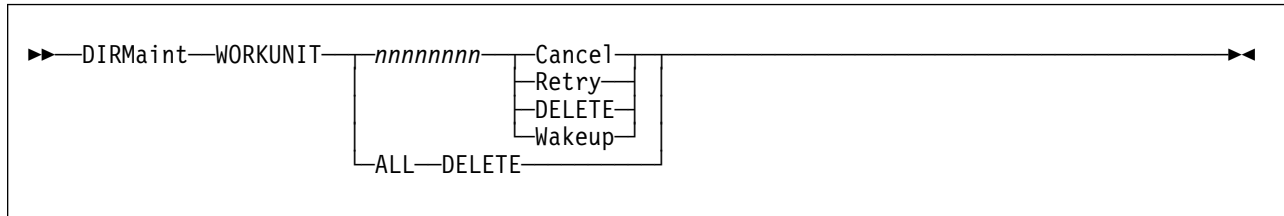
►►—DIRMaint—Verified—*vaddr*—————◄◄

VM Systems: All

Purpose

The VERIFIED operand of the DIRMAINT command is obsolete. This operand is tolerated to remain compatible with earlier releases of DirMaint. Its usage will not cause an error, but its function is obsolete. If issued, a warning message is returned indicating that the function of the operand is no longer available.

WORKUNIT



VM Systems: All

Purpose

The WORKUNIT operand of the DIRMAINT command is used to manipulate active and failed work units.

Operands

nnnnnnnn

is the identification number of the work unit.

Cancel

cancels the specified active work unit. This operand sends a cancel transaction to the DATAMOVE machine responsible for this work unit if the work unit is assigned. The cancel completes when the DATAMOVE machine responds. If the work unit is not assigned it is removed from the unassigned queue.

Wakeup

requests that a WAKEUP event be sent to the owner of the work unit (or all DATAMOVEs if this work unit is unassigned). This operand will cause DATAMOVE to examine its retry queue for any tasks that can be worked on. This is done automatically by DirMaint each time a DIRECT occurs.

DELETE

ALL DELETE

deletes the specified failed work unit file or "ALL" failed work unit files.

Note: The failed work unit file contains information that details how the failure occurred and what steps, if any, were taken by DirMaint to roll back the transaction.

Retry

requests that the DATAMOVE control command of an active work unit be retried if it is currently in status ACTIVE or RETRY.

Note: This operand redrives the DATAMOVE machine with the DMVCTL request. This operand may be useful if you suspect the DATAMOVE machine or some outside force has "purged the request" from the DATAMOVE transaction queue. If the status of the DMVCTL command is RETRY, this may indicate that the DATAMOVE machine is simply waiting for a DIRECT to occur or for all links to the target device to be released. If this is the case, you should use the WAKEUP operand to force DATAMOVE to reexamine its retry queues.

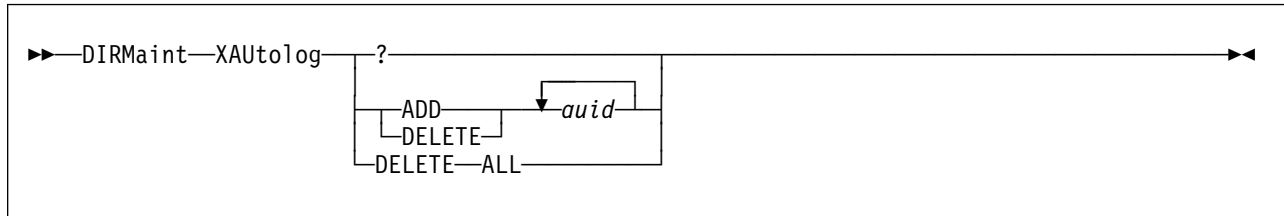
Usage Notes

1. You may use one or more of the prefix keywords (IMMed, REQuest, TOsys, ASuser, BYuser) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. A work unit is assigned and built to contain the sequence of commands and user actions required to execute certain complex commands. This mechanism is necessary because operands such as CMDISK could not be completed by a single command processor. In fact, they actually expand into a coordinated sequence of discrete events and are handled asynchronously. For example, the CMDISK operand expands to include allocation of a new extent, copying of data from the old minidisk to the new, confirmation of a successful copy, reformatting of the old minidisk, and release of the old extent.
3. A cancel or retry on a non-existent work unit is rejected with a message.
4. The CANCEL operand results in a CANCEL request being sent to the appropriate DATAMOVE machine (if the work unit is assigned). The DATAMOVE machine will respond with an appropriate CANCEL acknowledgement after it has cleaned up after the work element.

If the work element is not assigned (that is, the request had been accepted but no work has yet been done on it), it is simply deallocated.
5. The WAKEUP operand results in a WAKEUP request being sent to the appropriate DATAMOVE machine (if the work unit is assigned). The DATAMOVE machine will check its retry queue for work that can be done.

If the work unit is not currently assigned, a general WAKEUP request will be sent to all DATAMOVE machines not currently working on a task.
6. The RETRY operand results in the DATAMOVE control command being reissued if it was found to be in RETRY or ACTIVE status. If the DATAMOVE control command is not in one of these two states the command is rejected.

XAUTOLOG



VM Systems: VM/ESA (ESA Feature)

Purpose

The XAUTOLOG operand of the DIRMAINT command is used to add or delete a virtual machine name from an existing AUTOLOG or XAUTOLOG statement for a specified user's directory entry.

Operands

where:

? specifies that the existing AUTOLOG or XAUTOLOG statement is to be queried.

ADD

specifies that a user ID (*auid*) is to be added to an existing AUTOLOG or XAUTOLOG statement or a new statement is to be created.

DELETE

DELETE ALL

specifies that a user ID (*auid*) is to be deleted from the AUTOLOG or XAUTOLOG statement. If the ALL keyword is used the entire statement is deleted.

auid

specifies a 1 to 8 character name of the virtual machine authorized to AUTOLOG or XAUTOLOG the specified user ID.

Usage Notes

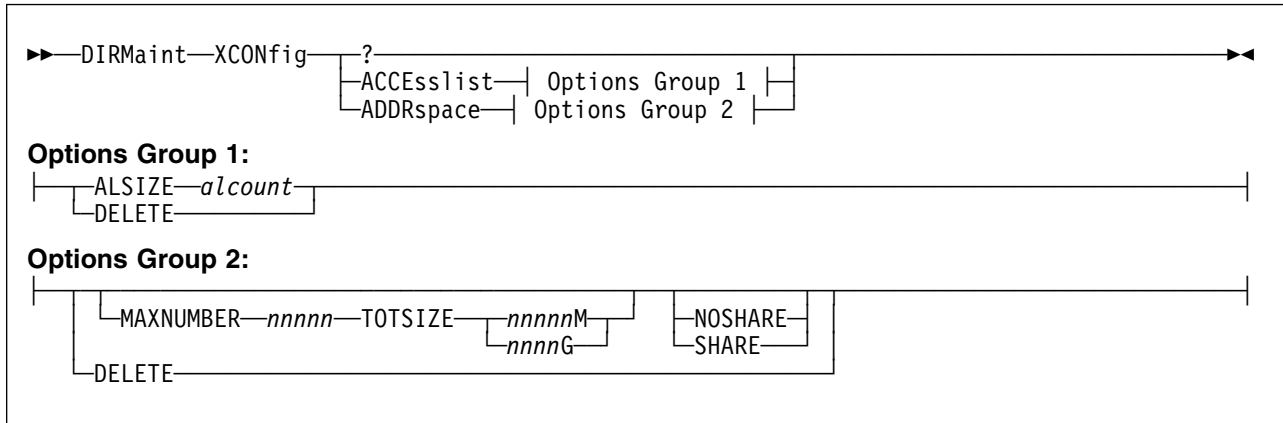
1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the AUTOLOG directory statement, see *z/VM: Planning and Administration*.
3. XAUTOLOG and AUTOLOG are synonymous but XAUTOLOG is the preferred directory keyword. Therefore, if the keyword AUTOLOG is encountered it is translated to XAUTOLOG.
4. A "DIRM XAUTOLOG ADD *auid*" may result in the creation of an XAUTOLOG statement, if there are no existing XAUTOLOG statements in the directory entry, or no more room on an existing XAUTOLOG statement. Similarly, a "DIRM XAUTOLOG DELETE *auid*" may result in the deletion of

XAUTOLOG

an XAUTOLOG statement if there are no more *auids* on the existing XAUTOLOG statement.

5. Specifying DELETE for a user ID not on an AUTOLOG or XAUTOLOG statement will generate an error message. Specifying ADD for a user ID already on an AUTOLOG or XAUTOLOG statement will also generate an error message.
6. A maximum of eight user IDs may be specified via AUTOLOG or XAUTOLOG.
7. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by a message that both the source and online directories have been updated.

XCONFIG



VM Systems: VM/ESA (ESA Feature)

Purpose

The XCONFIG operand of the DIRMAINT command allows authorized users to manipulate the XCONFIG directory statement. The XCONFIG statement specifies control parameters for the extended-configuration facilities provided in the XC virtual machine architecture.

Operands

? requests that the current settings of the XCONFIG statements be returned in a message.

ACCEsslist

specifies the size of the host access list to be provided for this virtual machine.

ADDRspace

specifies the maximum number of nonprimary address spaces and the total size in bytes of all nonprimary address spaces that the virtual machine can own simultaneously.

ALSIZE *alcounT*

specifies the size, in number of access-list entries, for this virtual machine's host access list.

MAXNUMBER *nnnnn*

specifies the maximum number of nonprimary address spaces that this virtual machine can create and have existing concurrently.

TOTSIZEnnnnnM

TOTSIZEnnnnG

specifies the maximum total size, in bytes, of all address spaces that this virtual machine can create and have existing concurrently.

SHARE

NOSHARE

specifies whether the virtual machine can or cannot use the PERMIT function of the ADRSPACE macroinstruction to make its address spaces available for access by other virtual machines.

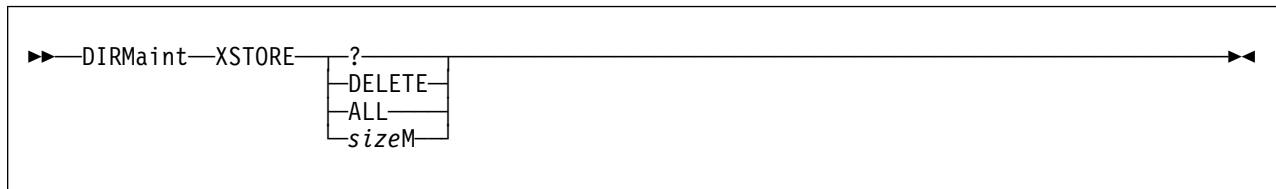
DELETE

requests that the specified statement be deleted from the directory.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the XCONFIG directory statement, see *z/VM: Planning and Administration*.

XSTORE



VM Systems: VM/ESA (ESA Feature)

Purpose

The XSTORE operand of the DIRMAINT command is used to authorize a virtual machine that supports Expanded Storage to use the Expanded Storage Facility.

Operands

? requests that the current setting of the XSTORE statement be displayed in the form of a message.

DELETE

specifies a request to delete the XSTORE statement from a directory entry.

ALL

specifies that all of the expanded storage configured on the system is allowed to be used for a guest partition.

sizeM

specifies the number of megabytes of real expanded storage to be dedicated to the user's virtual machine.

Usage Notes

1. You may use one or more of the prefix keywords (REQuest, TOsys, ASuser, BYuser, FORuser, ATnode) with this command. These prefix keywords may be specified in any order, but each keyword may only be specified once.
2. For more information on the XSTORE directory statement, see *z/VM: Planning and Administration*.
3. The directory change may be brought online immediately by the CP update-in-place facility (DIAGNOSE code X'84'). When this facility is used, you are informed by message that both the source and online directories have been updated.

Appendix A. DirMaint Command Sets

There are two files that define which command belongs to what command set(s): 140CMDS DATADVH and 150CMDS DATADVH. The command sets can be changed by your installation by editing these files and following the procedures documented in the files themselves.

The Table 1 summarizes the command set definitions for the DirMaint commands as they would be after initial installation and before any tailoring is done. The table also identifies if the user will be prompted for a password when they issue the command.

The IBM default command sets for DirMaint Release 5.0 are:

- A Administration, non-DASD related
- D DASD Management
- G General users
- H Helpdesk
- M Password Monitor
- O System Operator
- P DASD management automated Programs, such as DFSMS/VM
- S Support programmer
- Z Internal communication

Additional classes may be defined, but there is a maximum of 36 classes per command.

For comparison with DIRMAINT Release 4:

- DIRM_STAFF= classes ADHMP
- DIRM_SUBSTAFF= class H
- PWMON = class M
- OPERATOR= class O
- OWNER= class S

Table 1 (Page 1 of 6). DirMaint default command sets

| Command Name | 150A Command Set | | 140A Command Set | |
|---------------------|------------------|----|------------------|----|
| | Class(es) | PW | Class(es) | PW |
| ? | ..G..... | N | ..G..... | N |
| ACCOUNT | ..G..... | Y | ..G..... | Y |
| ACIGROUP | A..... | Y | | . |
| ACNTADD | ..G..... | Y | | . |
| ACNTDEL | ..G..... | Y | | . |
| ADD | A..... | Y | A..... | Y |
| AMDISK | .D..... | Y | .D..... | Y |
| APPCPASS | ..G..... | Y | ..G..... | Y |
| ASSIGN ¹ | | . | A.....S. | Y |
| AUTH ¹ | | . | ..G..... | Y |

¹ The ASSIGN and AUTH commands have been replaced by the AUTHFOR command.

Table 1 (Page 2 of 6). DirMaint default command sets

| Command Name | 150A Command Set | | 140A Command Set | |
|--|------------------|----|------------------|----|
| | Class(es) | PW | Class(es) | PW |
| AUTHBY | ..G..... | Y | | . |
| AUTHFOR | ..G..... | Y | | . |
| AUTHLINK | ..G..... | Y | | . |
| AUTHSCIF | ..G..... | Y | | . |
| AUTOLOG | ..G..... | Y | A..... | Y |
| BACKUP | ...H.O.S. | Y | ...H.O.S. | Y |
| BATCH | ..G..... | Y | ..G..... | Y |
| CHECK | ..G..... | N | | . |
| CHKSUM | ...H...S. | Y | A..H...S. | Y |
| CHNGID | A..... | Y | A..... | Y |
| CHVADDR | ..G..... | Y | A..... | Y |
| CLASS | A..... | Y | | . |
| CLEANED | .D..... | N | .D.....Z | Y |
| CMDISK | .D..... | Y | .D..... | Y |
| CMS |S. | Y |S. | Y |
| CONSOLE ² | ..G..... | Y | AD.H...S. | Y |
| <p>² In CMDLEVEL 140A, the CONSOLE command retrieves a copy of the DIRMAINT service machine's current console log. To obtain this function in CMDLEVEL 150A, use the GETCONSOLE command. In CMDLEVEL 150A, the CONSOLE command changes the virtual address or device type of a CONSOLE directory statement.</p> | | | | |
| COPIED | .D..... | N | .D.....Z | Y |
| CP |S. | Y |S. | Y |
| CPU | ..G..... | Y | ..G..... | Y |
| CRYPTO | A..... | Y | | . |
| DASDOPT | AD..... | Y | AD..... | Y |
| DATAMOVE |S. | Y | AD....S. | Y |
| DATEFORMAT |G. | Y | | . |
| DEDICATE |PS. | Y | | . |
| DEFAULTS | ..G..... | N | ..G..... | N |
| DEFINESTAG |PS. | Y | | . |
| DIRECT | AD.H..PS. | Y | AD....S. | Y |
| DIRECTORY |S. | Y | | . |
| DIREEDIT | AD.H..PS. | Y | AD.H...S. | Y |
| DIRMAP | AD.H..PS. | Y | | . |
| DISABLE |S. | Y |S. | Y |
| DISTRIB | ..G..... | Y | ..G..... | Y |
| DLINK | ..G..... | Y | ..G..... | Y |

Table 1 (Page 3 of 6). DirMaint default command sets

| Command Name | 150A Command Set | | 140A Command Set | |
|--|------------------|----|------------------|----|
| | Class(es) | PW | Class(es) | PW |
| DMDISK | .D..... | Y | .D..... | Y |
| DMVCTL |Z | Y | | . |
| DROP ³ | | . | A.....S. | Y |
| ³ The DROP and UNAUTH commands have been replaced by the DROPFOR command. | | | | |
| DROPHY | ..G..... | Y | | . |
| DROPFOR | ..G..... | Y | | . |
| DROPSCIF | ..G..... | Y | | . |
| DSATCTL |Z | Y | | . |
| DSECUSER | ..G..... | Y | | . |
| DUMP | ...H...S. | Y | ...H...S. | Y |
| DBONECMD | A..... | Y | | . |
| ELINK | A..... | Y | A..... | Y |
| ENABLE |PS. | Y |S. | Y |
| EXECDROP | ..G..... | N | | . |
| EXECLOAD | ..G..... | N | | . |
| EXTNCHK |PS. | Y | .D..... | Y |
| FILE |S. | Y |S. | Y |
| FREEXT | .D.H...S. | Y | .D.H...S. | Y |
| GET | A..... | Y | A..H.... | Y |
| GETCONSOLE | ...H...S. | Y | ...H...S. | Y |
| GLOBALOPTS |P.. | Y | | . |
| GLOBALV | ..G..... | N | ..G..... | N |
| HELP | ..G..... | N | ..G..... | N |
| HISTORY | ...H...S. | Y | ...H...S. | Y |
| INCLUDE | A..... | Y | | . |
| INVEN | ...H...S. | Y | ...H...S. | Y |
| IPL | ..G..... | Y | ..G..... | Y |
| IUCV | A..... | Y | | . |
| LINK | ..G..... | Y | ..G..... | Y |
| LOCK | A..... | Y | A..... | Y |
| LOGMSG | AD.H...S. | Y | AD.H...S. | Y |
| LOGONBY | ..G..... | Y | | . |
| MACHINE | ..G..... | Y | ..G..... | Y |
| MAIL | ...H...S. | Y | A.....S. | Y |
| MAXSPOOL | A..... | Y | A..... | Y |
| MAXSTOR | A..... | Y | A..... | Y |

Table 1 (Page 4 of 6). DirMaint default command sets

| Command Name | 150A Command Set | | 140A Command Set | |
|---|------------------|----|------------------|----|
| | Class(es) | PW | Class(es) | PW |
| MDAUDIT | AD..M.... | Y | AD..M.... | Y |
| MDISK | ..G..... | Y | ..G..... | Y |
| MDPW | ..G..... | Y | ..G..... | Y |
| MINIOPT | AD..... | Y | AD..... | Y |
| MMDISK | .D....P.. | Y | .D....P.. | Y |
| NAMESAVE | A..... | Y | A..... | Y |
| NEEDPASS | ..G..... | Y | ..G..... | Y |
| NEWS | ..G..... | N | ..G..... | N |
| NOPDATA | ..G..... | Y | ..G..... | Y |
| NOTAPE |0.S. | Y |0.S. | Y |
| OFFLINE | AD.....S. | Y | AD.....S. | Y |
| ONLINE | AD.....S. | Y | AD.....S. | Y |
| OPTION | ..G..... | Y | ..G..... | Y |
| POOL | A..... | Y | | . |
| POSIXFSROOT | ..G..... | Y | | . |
| POSIXGLIST | A..... | Y | | . |
| POSIXGROUP | A..... | Y | | . |
| POSIXINFO | A..... | Y | | . |
| POSIXIUPGM | ..G..... | Y | | . |
| POSIXIWDIR | ..G..... | Y | | . |
| POSIXOPT | A..... | Y | | . |
| PRIORITY | A..... | Y | | . |
| PRIOSET | A..... | Y | A..... | Y |
| PRIVCLAS | | . | A..... | Y |
| PRIVCLASS | A..... | Y | | . |
| PURGE | A..... | Y | A..... | Y |
| PW | ..G..... | Y | ..G..... | Y |
| PW? ⁴ | ..G..... | N | ..G..... | N |
| <p>⁴ When no administratively controlled password change is pending, you are not normally prompted to enter a password for the PW? command. However, if prompted, your entry will not be verified. When an administratively controlled password change is pending, you will be prompted to enter a password that will be verified before the system responds with your new password.</p> | | | | |
| PWGEN |M.... | Y |M..S. | Y |
| PWMON |M.... | Y |M..S. | Y |
| QLOG | ..G..... | Y | ..G..... | Y |
| QRY | AD.H...S. | Y | AD.H...S. | Y |
| QUERY | AD.H...S. | Y | AD.H...S. | Y |

Table 1 (Page 5 of 6). DirMaint default command sets

| Command Name | 150A Command Set | | 140A Command Set | |
|--|------------------|----|------------------|----|
| | Class(es) | PW | Class(es) | PW |
| REPLACE |S. | Y | A.....S. | Y |
| REVIEW | ..G..... | Y | ..G..... | Y |
| RLDCODE |S. | Y |S. | Y |
| RLDDATA |S. | Y |S. | Y |
| RLDEXTN |S. | Y |S. | Y |
| RMDISK | .D....P.. | Y | .D....P.. | Y |
| SATELLITE |S. | Y | | . |
| SCAN | ...H...S. | Y | A..H.... | Y |
| SCREEN | ..G..... | Y | ..G..... | Y |
| SECUSER | ..G..... | Y | | . |
| SEND | ...H...S. | Y |S. | Y |
| SETACNT | A..... | Y | | . |
| SETCLASS | A..... | Y | | . |
| SETCPU | A..... | Y | A..... | Y |
| SETMACH | A..... | Y | A..... | Y |
| SETOPTN ⁵ | A..... | Y | ..G..... | Y |
| <p>⁵ In CMDLEVEL 140A, the SETOPTN command manipulates the user's DirMaint control options. To obtain this function in CMDLEVEL 150A, use the USEROPTN command. In CMDLEVEL 150A, the SETOPTN command allows an administrator to add or change values on a user's OPTION directory statement.</p> | | | | |
| SETPRIORITY | A..... | Y | | . |
| SETPW | A...M.... | Y | A...M.... | Y |
| SETSTAG | A..... | Y | | . |
| SHARE | A..... | Y | | . |
| SHUTDOWN |O.S. | Y |O.S. | Y |
| SPECIAL | A..... | Y | | . |
| SPOOL | ..G..... | Y | | . |
| STAG | ..G..... | Y | | . |
| STATUS | ...H...S. | Y | ...H...S. | Y |
| STDEVOPT | A..... | Y | | . |
| STORAGE | ..G..... | Y | ..G..... | Y |
| SUPGLIST | ..G..... | Y | | . |
| SYSAFFIN | A..... | Y | A..... | Y |
| TERM | ..G..... | Y | ..G..... | Y |
| TESTPW | ..G..... | Y | ..G..... | Y |
| TMDISK | A..... | Y | A..... | Y |
| UNAUTH ⁶ | | . | ..G..... | Y |

Table 1 (Page 6 of 6). DirMaint default command sets

| Command Name | 150A Command Set | | 140A Command Set | |
|---|------------------|----|------------------|----|
| | Class(es) | PW | Class(es) | PW |
| 6 The UNAUTH and DROP commands have been replaced by the DROPFOR command. | | | | |
| UNLOCK | A..... | Y | A..... | Y |
| USEDEXT | .D.H...S. | Y | ...H...S. | Y |
| USER | ...H...S. | Y | ...H...S. | Y |
| USERMAP | ...H...S. | Y | ...H...S. | Y |
| USEROPTN | ..G..... | Y | ..G..... | Y |
| VERIFIED | ..G..... | N | ..G..... | Y |
| WORKUNIT | .D..... | Y | AD..... | Y |
| XAUTOLOG | ..G..... | Y | | . |
| XCONFIG | A..... | Y | | . |
| XSTORE | A..... | Y | A..... | Y |
| ZAPMDISK |Z | Y | | . |

Appendix B. Device Characteristics

Maximum Minidisk Sizes and Blocks Per Cylinder for CKD and FBA

Table 2 (Page 1 of 2). Maximum Minidisk Sizes and Blocks Per Cylinder for CKD

| Disk Type | Maximum CKD Minidisk Sizes | | Blocks Per CKD Cylinder with a Block Size of: | | | | |
|-----------|----------------------------|-------------|---|-----|-----|-----|-----|
| | 800-Byte CMS Format | Other Usage | 800 | 512 | 1K | 2k | 4K |
| 2305-1 | N/S | 48 | N/S | N/S | N/S | N/S | N/S |
| 2305-2 | N/S | 96 | N/S | N/S | N/S | N/S | N/S |
| 231x | 203 | 203 | 150 | 220 | 120 | 60 | 30 |
| 3330-1 | 246 | 404 | 266 | 380 | 209 | 114 | 57 |
| 3330-11 | 246 | 808 | 266 | 380 | 209 | 114 | 57 |
| 3340-35 | 348 | 348 | 96 | 144 | 84 | 42 | 24 |
| 3340-70 | 682 | 696 | 96 | 144 | 84 | 42 | 24 |
| 3350 | 115 | 555 | 570 | 810 | 450 | 240 | 120 |
| 3375 | 182 | 959 | 360 | 480 | 300 | 168 | 96 |
| 3380-1 | 121 | 885 | 540 | 690 | 465 | 270 | 150 |
| 3380-2 | 121 | 1770 | 540 | 690 | 465 | 270 | 150 |
| 3380-3 | 121 | 2655 | 540 | 690 | 465 | 270 | 150 |
| 3380-91 | N/S | 1113 | 540 | 690 | 465 | 270 | 150 |
| 3380-92 | N/S | 2226 | 540 | 690 | 465 | 270 | 150 |
| 3380-93 | N/S | 3339 | 540 | 690 | 465 | 270 | 150 |
| 3380-99 | N/S | 10017 | 540 | 690 | 465 | 270 | 150 |
| 3380-459 | N/S | 1459 | 540 | 690 | 465 | 270 | 150 |
| 3390-1 | N/S | 1113 | N/S | 735 | 495 | 315 | 180 |
| 3390-2 | N/S | 2226 | N/S | 735 | 495 | 315 | 180 |
| 3390-3 | N/S | 3339 | N/S | 735 | 495 | 315 | 180 |
| 3390-9 | N/S | 10017 | N/S | 735 | 495 | 315 | 180 |
| 3390-084 | N/S | 1084 | N/S | 735 | 495 | 315 | 180 |
| 3390-151 | N/S | 2226 | N/S | 735 | 495 | 315 | 180 |
| 3390-153 | N/S | 4365 | N/S | 735 | 495 | 315 | 180 |
| 3390-455 | N/S | 455 | N/S | 735 | 495 | 315 | 180 |
| 3390-568 | N/S | 1568 | N/S | 735 | 495 | 315 | 180 |
| 9345-1 | N/S | 1440 | N/S | 615 | 420 | 255 | 150 |

Table 2 (Page 2 of 2). Maximum Minidisk Sizes and Blocks Per Cylinder for CKD

| Maximum CKD Minidisk Sizes | | | Blocks Per CKD Cylinder with a Block Size of: | | | | |
|----------------------------|---------------------|-------------|---|-----|-----|-----|-----|
| Disk Type | 800-Byte CMS Format | Other Usage | 800 | 512 | 1K | 2k | 4K |
| 9345-2 | N/S | 2156 | N/S | 615 | 420 | 255 | 150 |

Special Notes:

1. N/S = NOT supported.
2. The 2305 is not supported by CMS at any block size.
3. The 231x devices (2311/2314/2319) are no longer supported. Capacity information is shown above for migration planning.
4. For a 3333, specify 3330. For a 3344, specify 3340.
5. For a 3350 used in 3330 compatibility mode, specify 3330-1.
6. For a 3390 used in 3380 compatibility mode, specify:

| Disk Type | Cylinders |
|----------------|-----------|
| 3380-91 | 1113 |
| 3380-92 | 2226 |
| 3380-93 | 3339 |
| 3380-99 | 10017 |

7. For a 9392, specify 3390.
8. For a 9395, specify either 3390-3 (native) or 3380-3 (emulation).
9. For a 3995-151, specify 3390-151. For a 3995-153, specify 3390-153.
10. For a Multiprise® 2000 internal disk (or any other DASD) emulating a 3380:

| Disk Type | Cylinders |
|-----------------|-----------|
| 3380-01 | 885 |
| 3380-02 | 1770 |
| 3380-03 | 2655 |
| 3380-459 | 1459 |

11. For a Multiprise 2000 internal disk (or any other DASD) emulating a 3390:

| Disk Type | Cylinders |
|-----------------|-----------|
| 3390-01 | 1113 |
| 3390-02 | 2226 |
| 3390-03 | 3339 |
| 3390-09 | 10017 |
| 3390-455 | 455 |
| 3390-568 | 1568 |

12. For any other DASD that emulates a VM-supported disk device with a non-standard capacity, assign your own *model number*. Simply update the AUTOBLOCK and DEFAULTS sections of the EXTENT CONTROL file to include your new *model(s)* of the emulated device type.

Table 3 (Page 1 of 2). Maximum Minidisk Sizes and Blocks Per Cylinder for FBA

| Maximum CKD Minidisk Sizes | | | Blocks Per CKD Cylinder with a Block Size of: | | | | |
|----------------------------|---------------------|-------------|---|-----|----|----|----|
| Disk Type | 800-Byte CMS Format | Other Usage | 800 | 512 | 1K | 2k | 4K |
| 3310 | N/S | 126016 | N/S | 1 | 2 | 4 | 8 |
| 3370-1 | N/S | 558000 | N/S | 1 | 2 | 4 | 8 |
| 3370-2 | N/S | 712752 | N/S | 1 | 2 | 4 | 8 |
| 9313 | N/S | 246240 | N/S | 1 | 2 | 4 | 8 |
| 9332-400 | N/S | 360036 | N/S | 1 | 2 | 4 | 8 |

Table 3 (Page 2 of 2). Maximum Minidisk Sizes and Blocks Per Cylinder for FBA

| Maximum CKD Minidisk Sizes | | | Blocks Per CKD Cylinder with a Block Size of: | | | | |
|----------------------------|---------------------|-------------|---|-----|----|----|----|
| Disk Type | 800-Byte CMS Format | Other Usage | 800 | 512 | 1K | 2k | 4K |
| 9332-600 | N/S | 554800 | N/S | 1 | 2 | 4 | 8 |
| 9335 | N/S | 804714 | N/S | 1 | 2 | 4 | 8 |
| 9336-010 | N/S | 920115 | N/S | 1 | 2 | 4 | 8 |
| 9336-020 | N/S | 1672881 | N/S | 1 | 2 | 4 | 8 |
| 0671-00 | N/S | 574560 | N/S | 1 | 2 | 4 | 8 |
| 0671-04 | N/S | 624456 | N/S | 1 | 2 | 4 | 8 |
| 0671-08 | N/S | 513072 | N/S | 1 | 2 | 4 | 8 |
| V-DISK | N/S | 4194296 | N/S | 1 | 2 | 4 | 8 |

Special Notes:

1. N/S = NOT supported.
2. The 0671 DASD is also known as a 9371.
3. When used on a VM/ESA 370 feature system, all of the above device types must be specified as FB-512. When used on a VM/ESA ESA feature system, the above device types may be specified as either the actual device type or as FB-512.
4. For a virtual disk in storage (V-DISK), the device type must be specified as FB-512.
5. CP allocates virtual disks in storage and FBA disks in 8-block pages. Therefore, the size of the V-DISK or FBA disk may be rounded up to the nearest page.

Appendix C. The Synchronous Application Programming Interface (SAPI)

GUI

The Synchronous Application Programming Interface (SAPI) provides a method for exec programs to parse message responses from DirMaint commands. When this interface is activated, key information from the message response is returned in a structured and predictable format that allows a program to parse the fields of the response, analyze the information, and take any needed action.

Sample Programs

The DIRMSAPI EXEC is an example of a REXX program that sets up the proper environment for using the SAPI interface. The DVHSAPI EXEC issues a command to the DIRMAINT service machine and waits for the response, which it passes back to DIRMSAPI EXEC. Both of these sample exec files reside on the user interface disk (P748XE4M 11F, by default) as file names DIRMSAPI EXEC SAMP and DVHSAPI EXEC SAMP. These files are available for your examination.

Given a DirMaint command string, the DIRMSAPI EXEC routine calls the DVHSAPI EXEC to send the command to the DIRMAINT service machine and wait for the responses from the service machine. The responses are returned from DVHSAPI in the "DVHSAPI." stem variable.

The DVHSAPI EXEC routine is not intended to be invoked as a command directly from the console. It is intended to be called by a customer supplied REXX program. The DIRMSAPI EXEC (or DIRMSAPI EXEC SAMP file) is a sample of such a program.

Applied SAPI Coding Rules

The following rules have been applied to the DIRMSAPI EXEC sample.

- An EXEC DIRMAINT EXECLOAD command should be issued before calling DVHSAPI, and an EXEC DIRMAINT EXECDROP command should be issued when no further calls to DVHSAPI are anticipated for a significant time interval. When issuing multiple commands, a single EXEC DIRMAINT EXECLOAD command should be used before the group of calls to DVHSAPI, and one EXEC DIRMAINT EXECDROP command should be used after the group. Service machines that frequently issue DIRMAINT commands should include an EXEC DIRMAINT EXECLOAD command in their PROFILE EXEC.
- If the PRESET global variable, in the DVH15 pool, has a value other than blanks or NOLOG, DirMaint will send that value to the DIRMAINT server without prompting for a password. To prevent disclosure of that password, the variable will be reset to a value of NOLOG upon exit, thus the value must be set prior to each call to the DIRMAINT EXEC.

Note: If an EXEC DIRMAINT EXECLOAD command has already been issued and the files are still loaded, it is prudent to issue an EXEC DIRMAINT EXECDROP command before re-issuing the EXEC DIRMAINT EXECLOAD command. It does no harm to always issue an EXEC DIRMAINT EXECDROP command before issuing any EXEC DIRMAINT EXECLOAD command.

```

/* Miniature sample for use of DVHSAPI. */
Address 'COMMAND'
Trace 'C'
'EXEC DIRMAINT EXECLOAD'
'EXEC DVHSAPI FOR USER1 DISTRIB NEWDIST1'
'PIPE STEM DVHSAPI. | CONSOLE'
'EXEC DVHSAPI FOR USER2 DISTRIB NEWDIST2'
'PIPE STEM DVHSAPI. | CONSOLE'
'EXEC DVHSAPI FOR USER3 DISTRIB NEWDIST3'
'PIPE STEM DVHSAPI. | CONSOLE'
'EXEC DVHSAPI FOR USER4 DISTRIB NEWDIST4'
'PIPE STEM DVHSAPI. | CONSOLE'
'EXEC DIRMAINT EXECDROP'
Exit

```

- Calls to DVHSAPI may optionally preface the request with a time-out limit between 10 seconds (+00:00:1) and just under 12 hours (+11:59:5). The default is 15 seconds (+00:15), if not specified.
- The return code from DVHSAPI indicates the status of your request.
 - 1191** Message DVH1191I was expected but not received. Your command has not been sent to the DIRMAINT service machine. The messages passed back in the DVHSAPI stem variable explain why.
 - 2288** Message DVH2288I was expected but not received. The DIRMAINT service machine received your request, but has rejected it from further processing. The messages passed back in the DVHSAPI stem variable explain why.
 - 2289** Message DVH2289E was received. The DIRMAINT service machine processed your request, which ended with a non-zero return code. The messages passed back in the DVHSAPI stem variable explain why.
 - 0** Message DVH2289I was received. The DIRMAINT service machine processed your request, which ended with a zero return code. The DVHSAPI stem variable will contain the messages, if any, generated while processing your request.
 - xxxx** Other non-zero return codes indicate one of three things:
 1. The DIRMAINT service machine failed to respond to your request within the allotted time interval; or
 2. The DVHSAPI EXEC was confused by messages from the DIRMAINT service machine generated by a previous request, or by spontaneous messages from the DIRMAINT service machine; or
 3. Some other error condition was encountered that the DVHSAPI EXEC could not handle.
- Be aware that some commands result in asynchronous processing. For example, an EXEC DVHSAPI +00:05:0 BATCH SETALL MYPWS command will send the contents of the SETALL MYPWS file to the DIRMAINT service machine for processing, with message DVH2289I and a zero return code indicating that the file has been received and appended to the batch queue for processing. A zero return code from DVHSAPI in this case does not indicate that any or all of the requests within the file have been processed, or that they will all be processed successfully. For example, if one of the records in the file is FOR DVHTEST PW DVHTEST, this request will very likely fail with message

DVH2924E, even though you have already received message DVH2289I and a zero return code from your BATCH command.

- The DVHSAPI_END_MSG.CONFIG* DATADVH entry can be used to identify when the DVHSAPI routine exits and returns control back to the calling application. The default is to end when message DVHREQ2289I is received and no DVHSAPI_END_MSG. entries are specified. The sample CONFIG SAMPDVH file contains these entries:

- DVHSAPI_END_MSG.DVHSCU3541I= DVHREQ2289I DVHSHN3430I
- DVHSAPI_END_MSG.DVHPUR3370I= DVHREQ2289I DVHDMD3525I

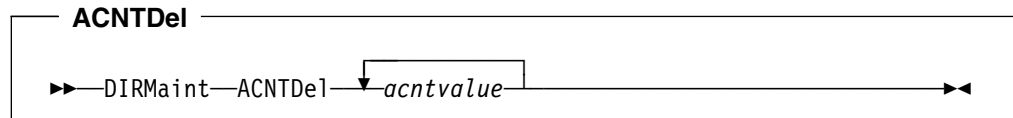
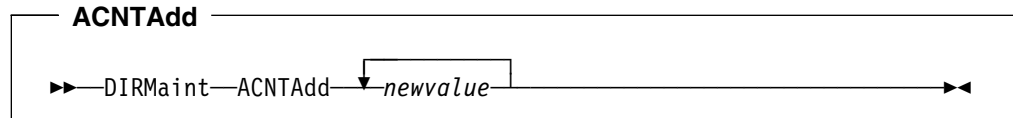
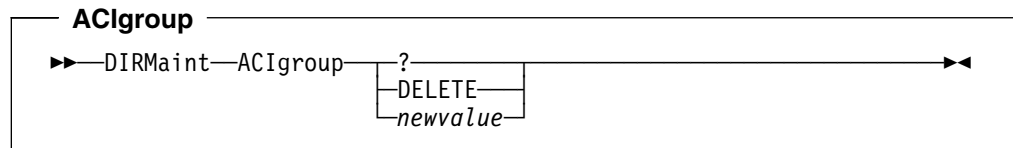
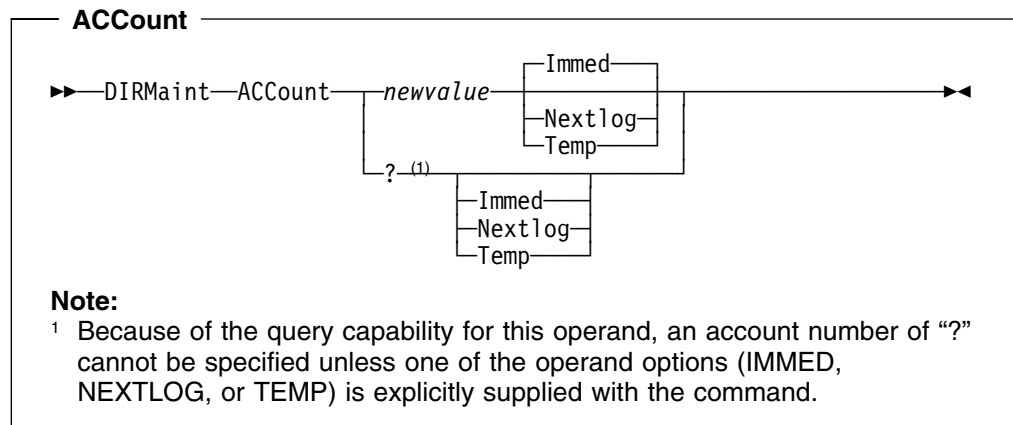
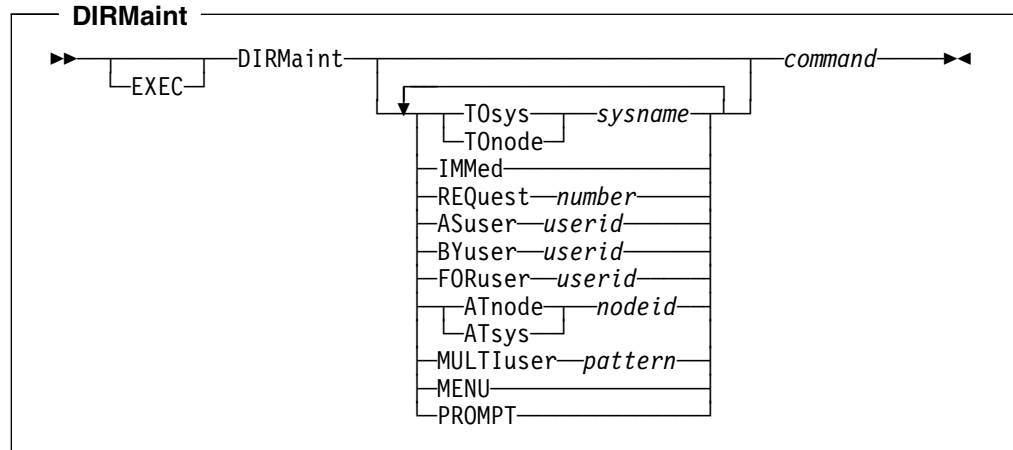
The message DVHSAPI_END_MSG.DVHSCU3541I= DVHREQ2289I DVHSHN3430I tells DVHSAPI that if it receives the message DVHSCU3541I, it should wait for both message DVHREQ2289I and DVHSHN3430I to be received, in any order, before determining whether the command finished successfully.

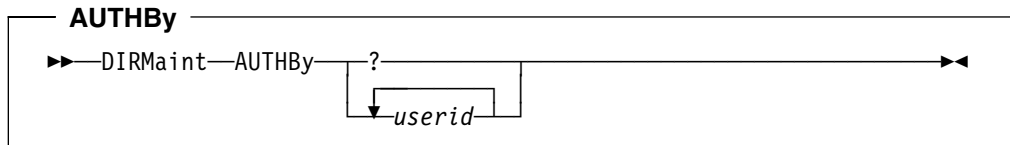
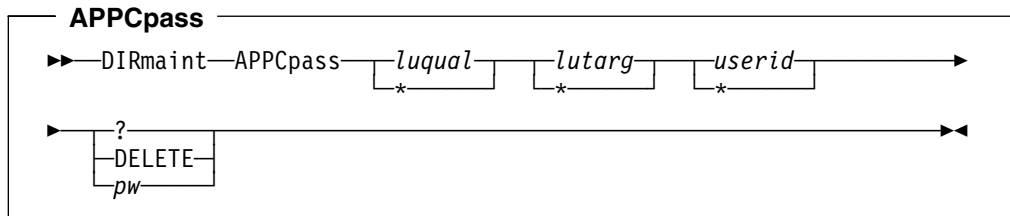
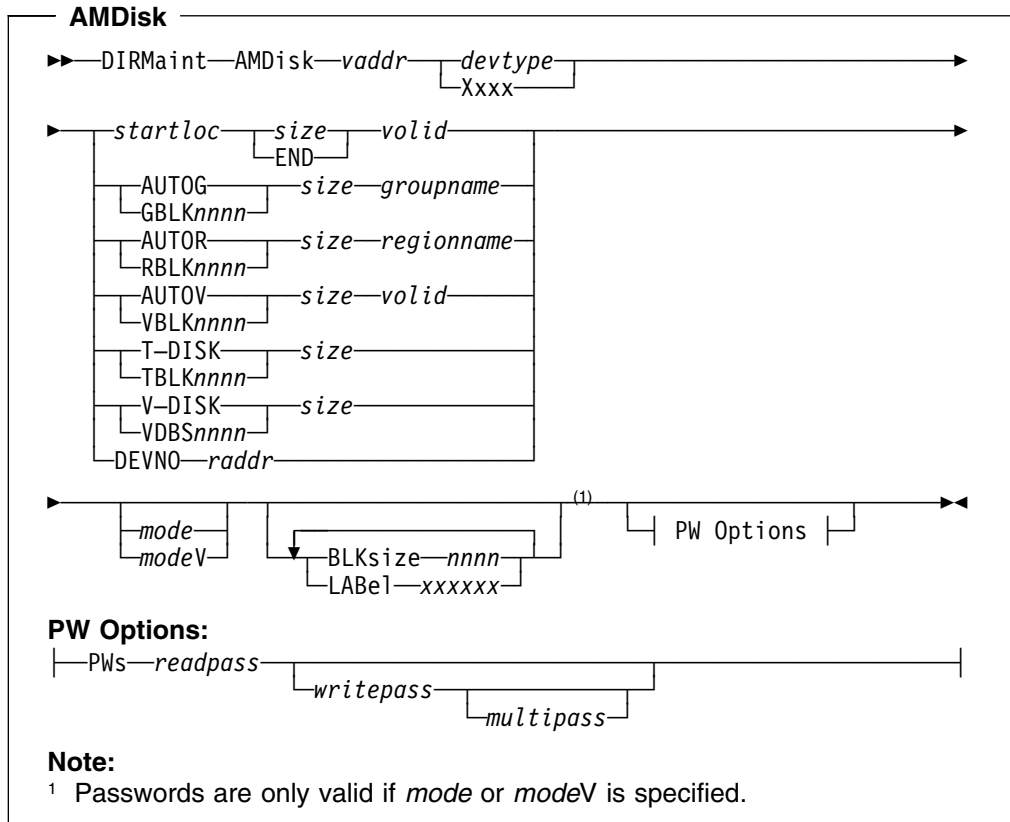
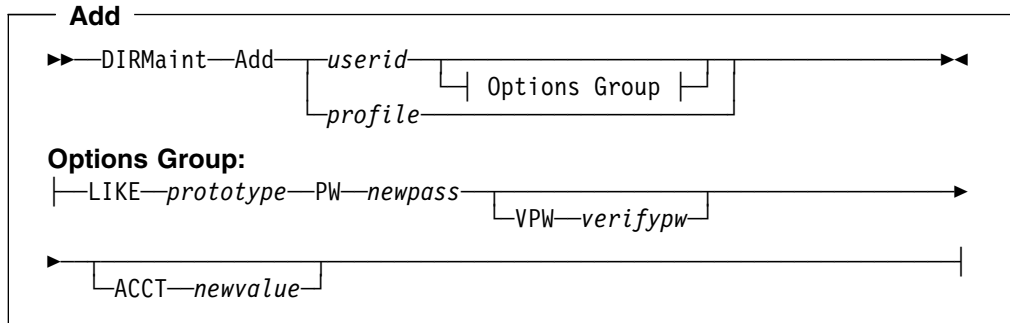
Note: This configuration entry can be used for any other condition requiring multiple messages to be received when determining if the command completed successfully.

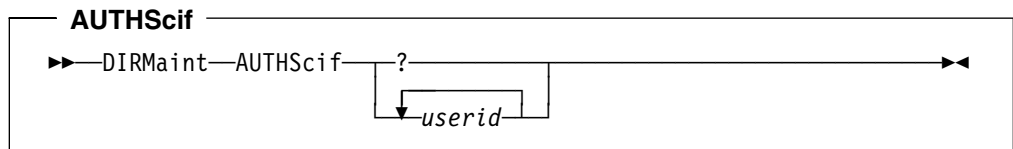
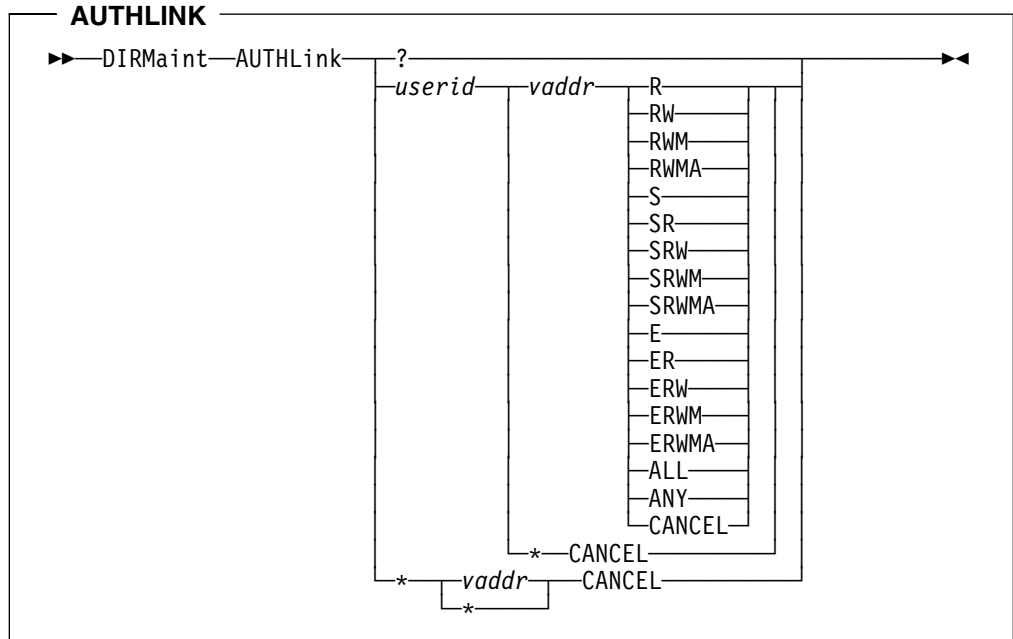
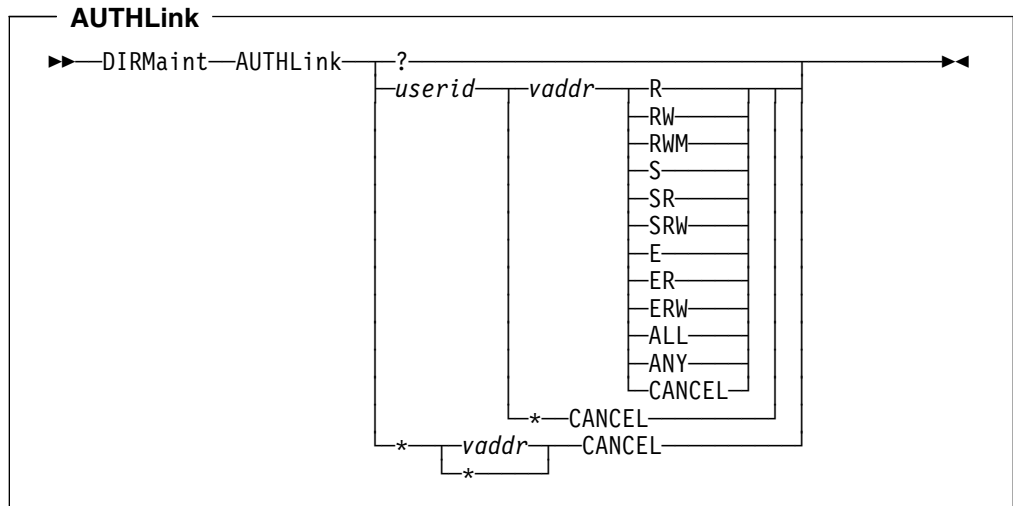
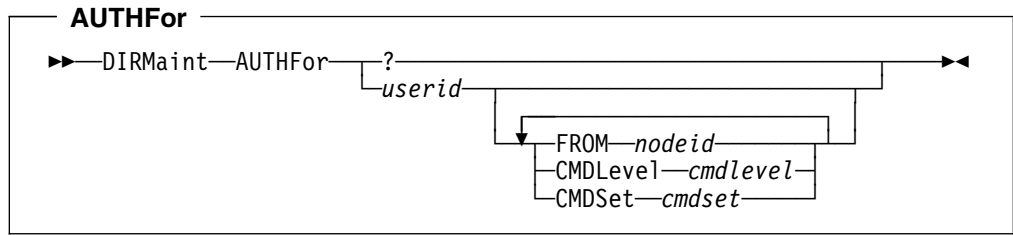
- The DVHSAPI EXEC does not support use of the TOsys or TOnode prefix operand for remote networking. When used within a multi-system CSE cluster, cross-system SMSG capability must be enabled.

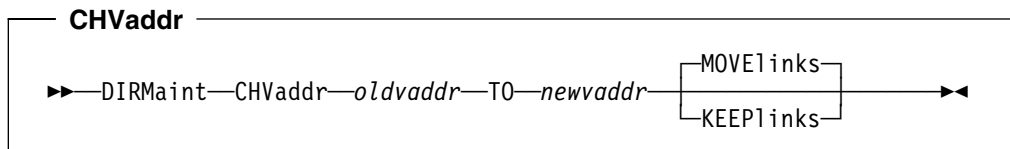
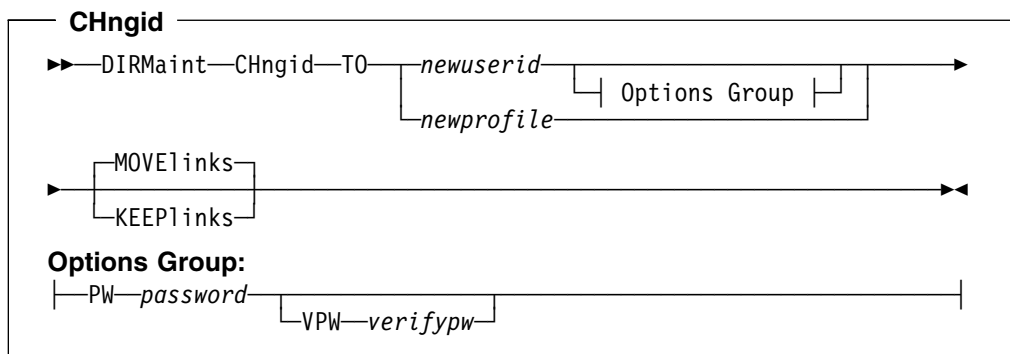
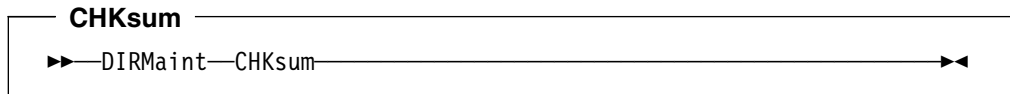
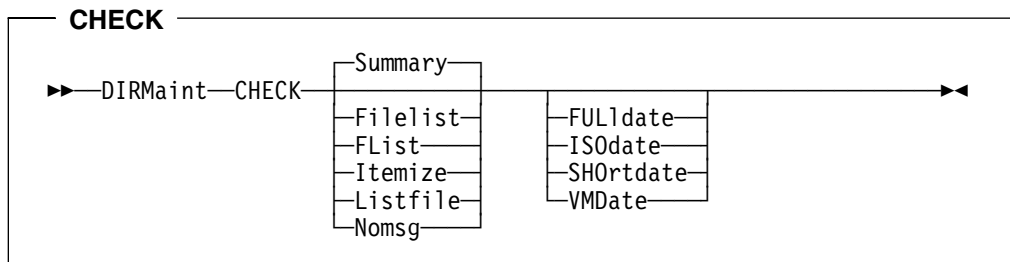
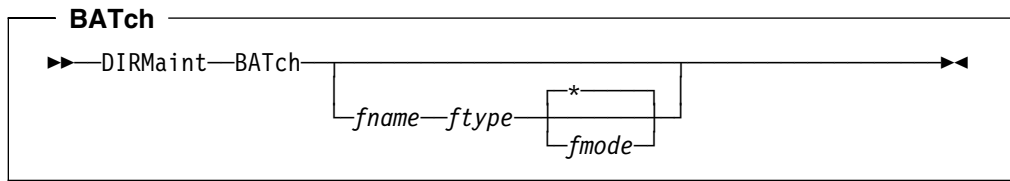
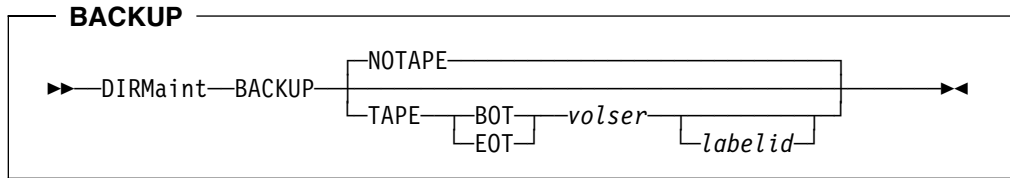
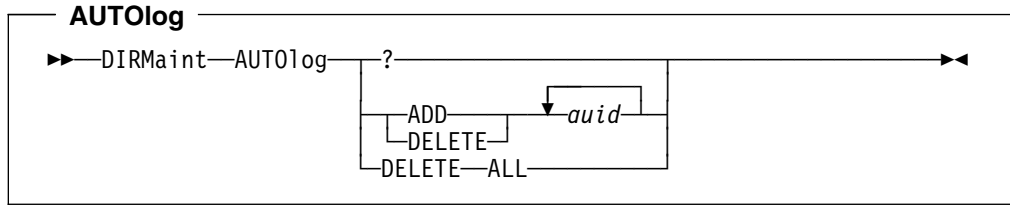
GUI end

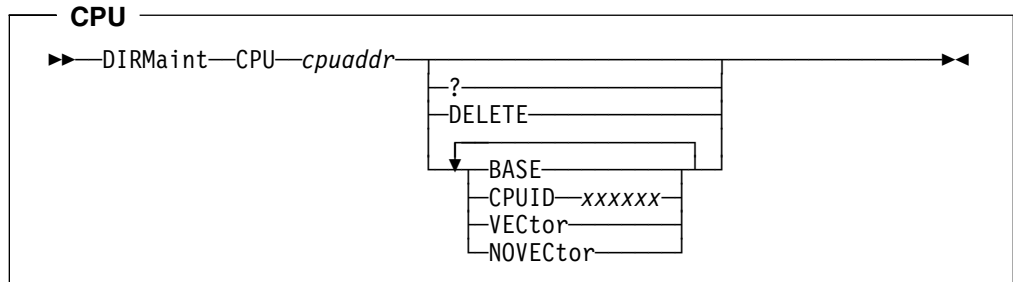
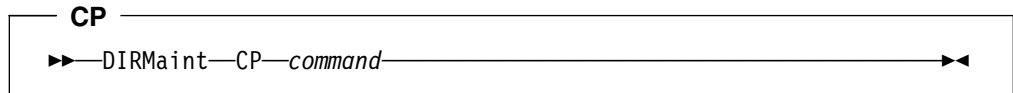
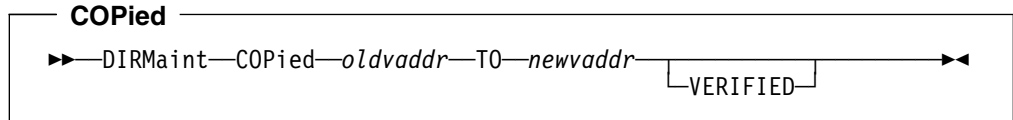
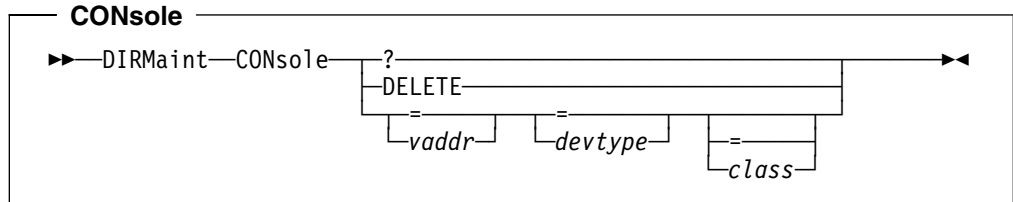
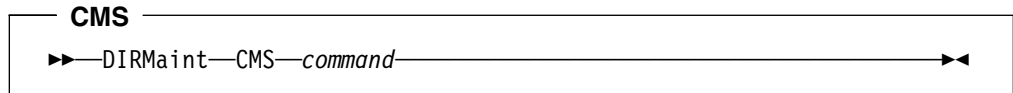
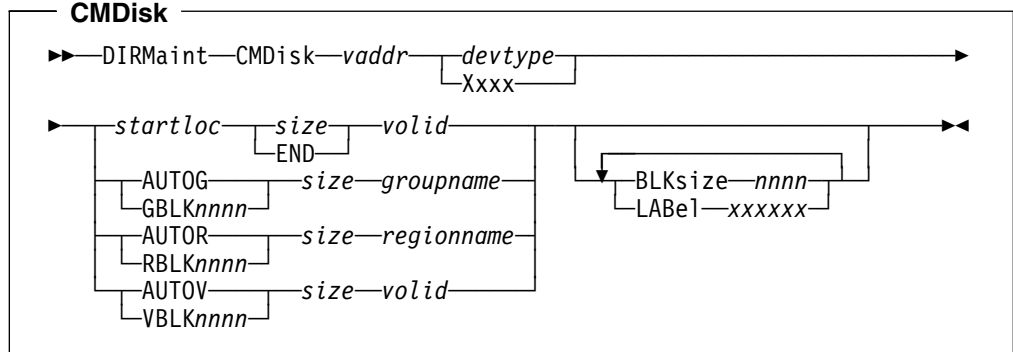
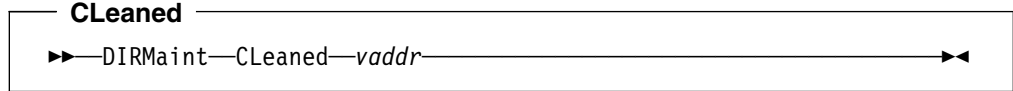
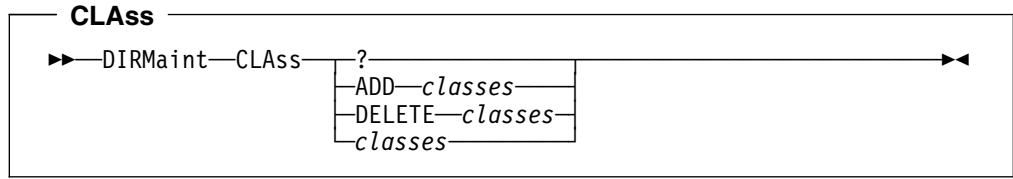
Appendix D. DirMaint Command Summary

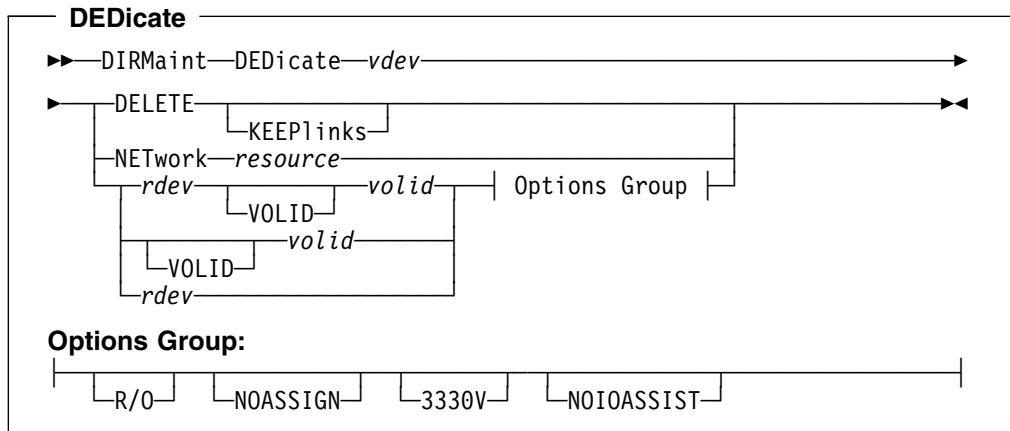
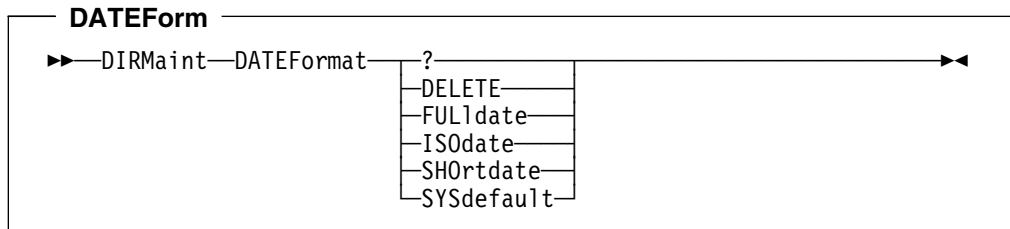
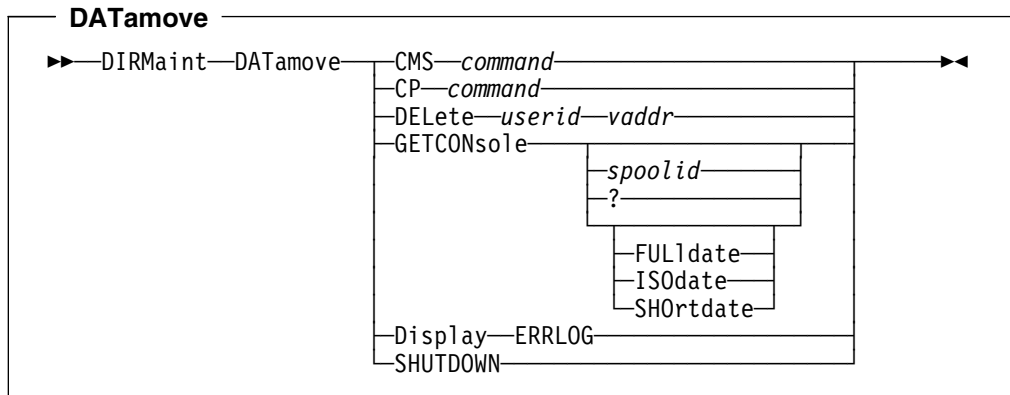
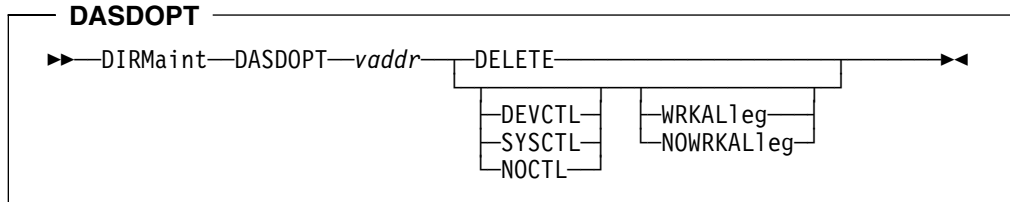
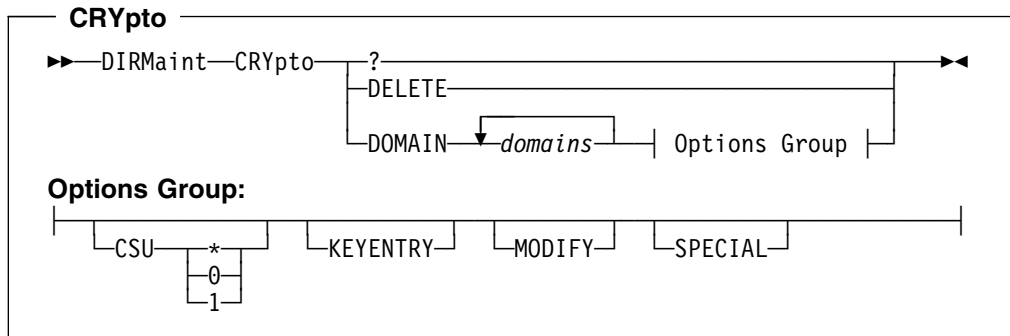




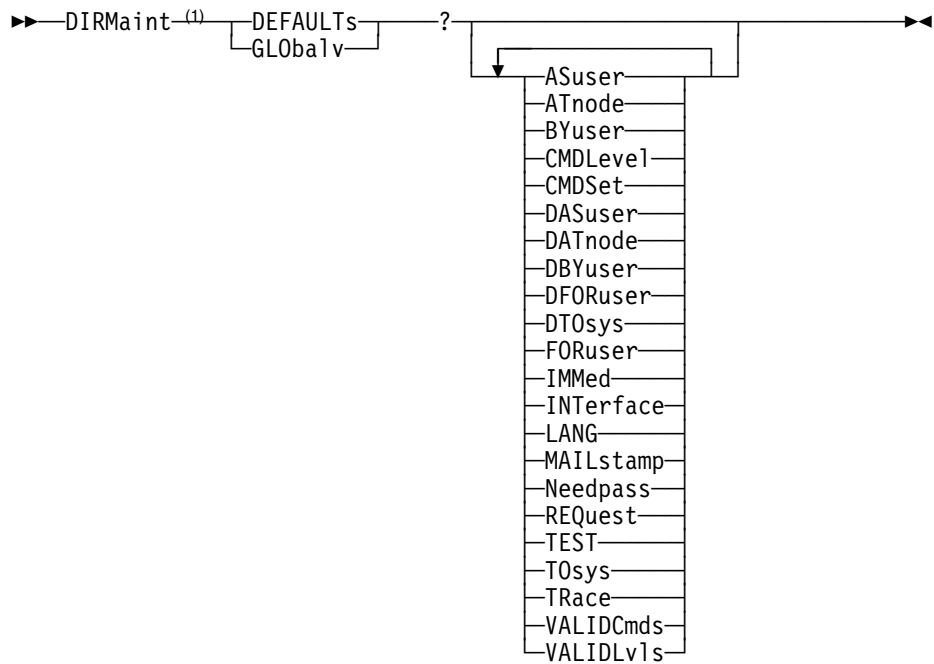








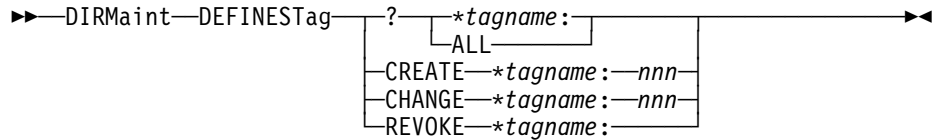
DEFAULTS, GLObalv



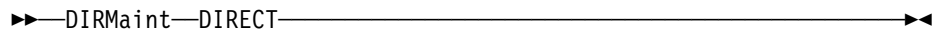
Note:

¹ Use the above command syntax to determine the existing value of a DirMaint GLOBALV variable.

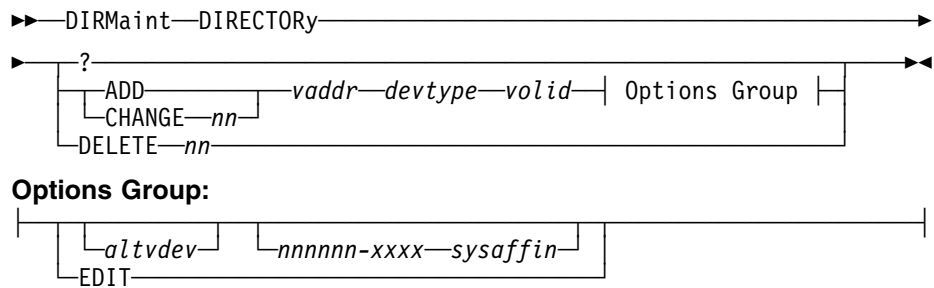
DEFINESTag

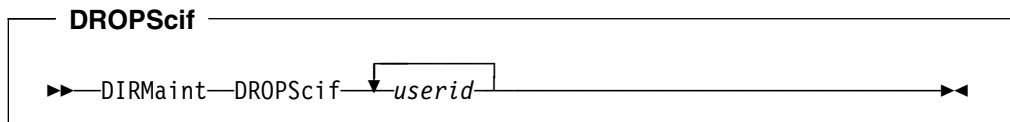
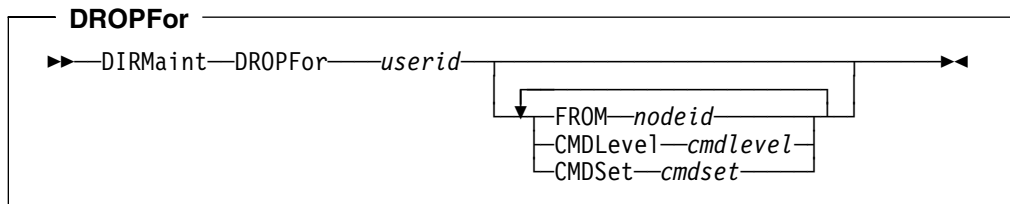
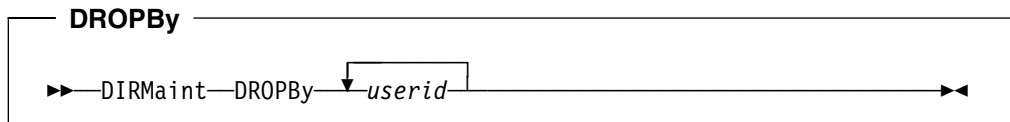
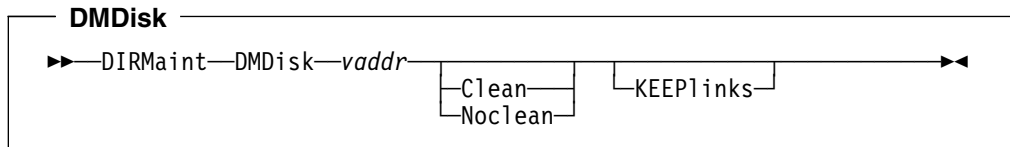
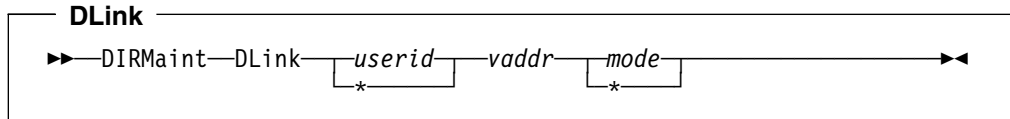
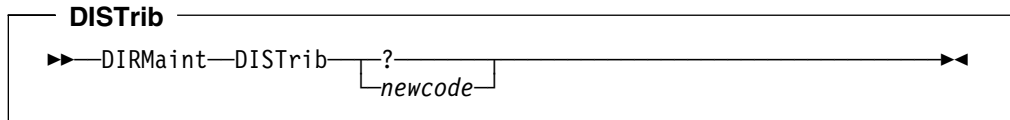
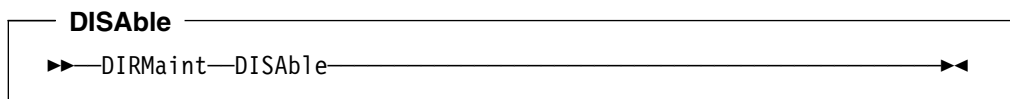
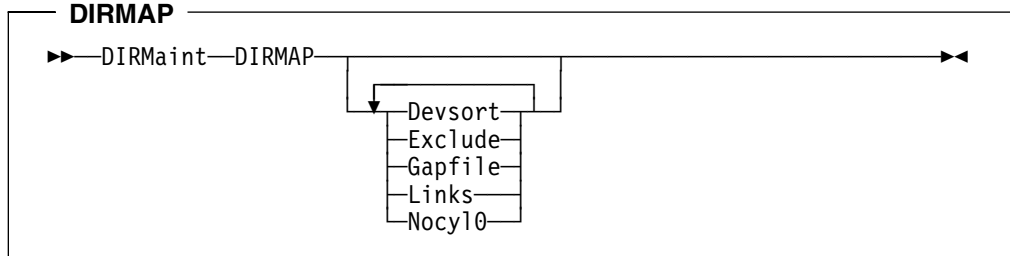
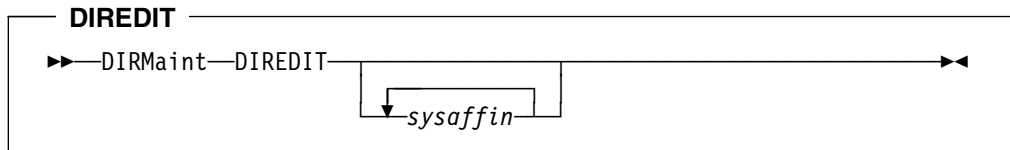


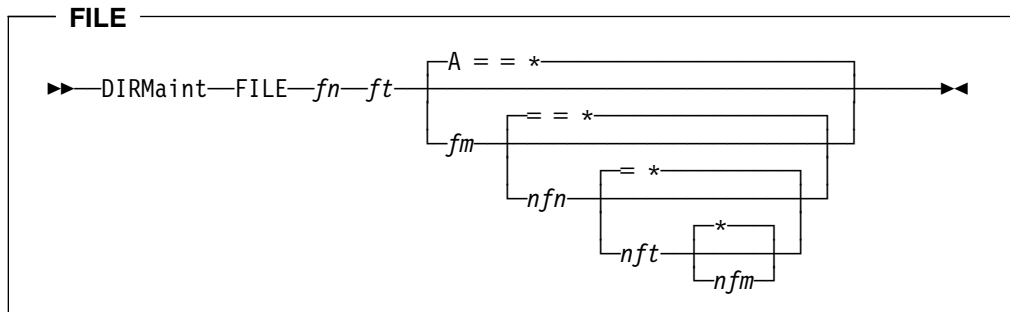
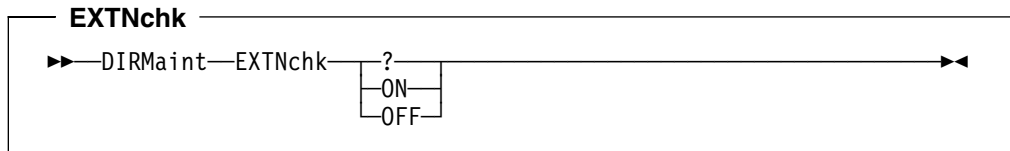
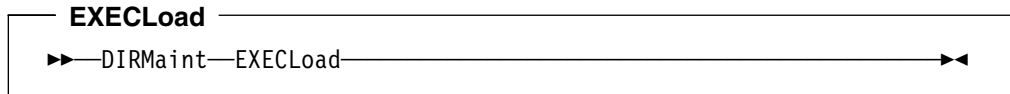
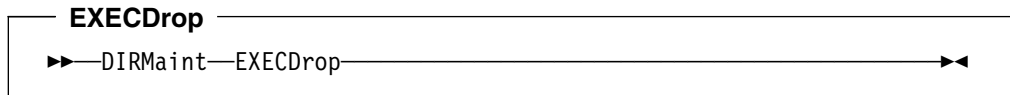
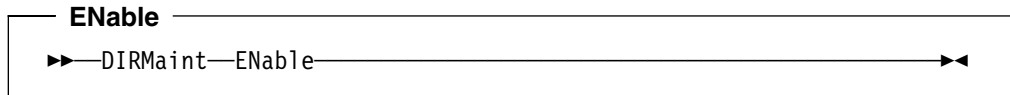
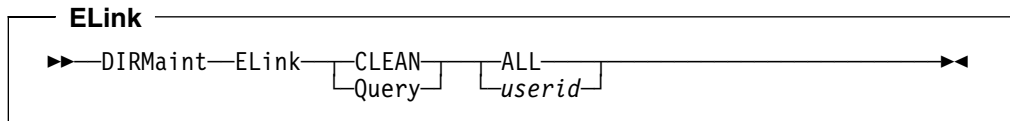
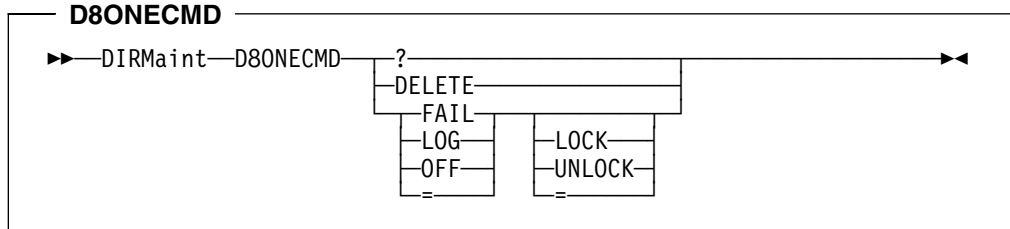
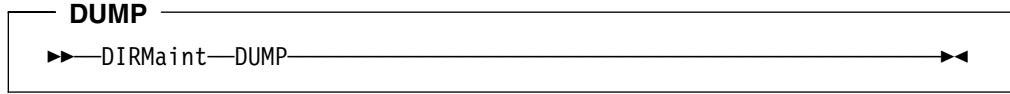
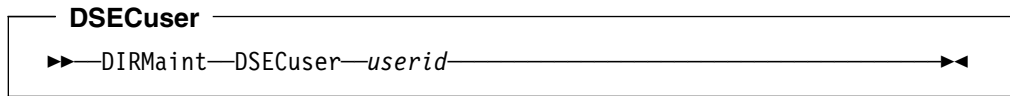
DIRECT

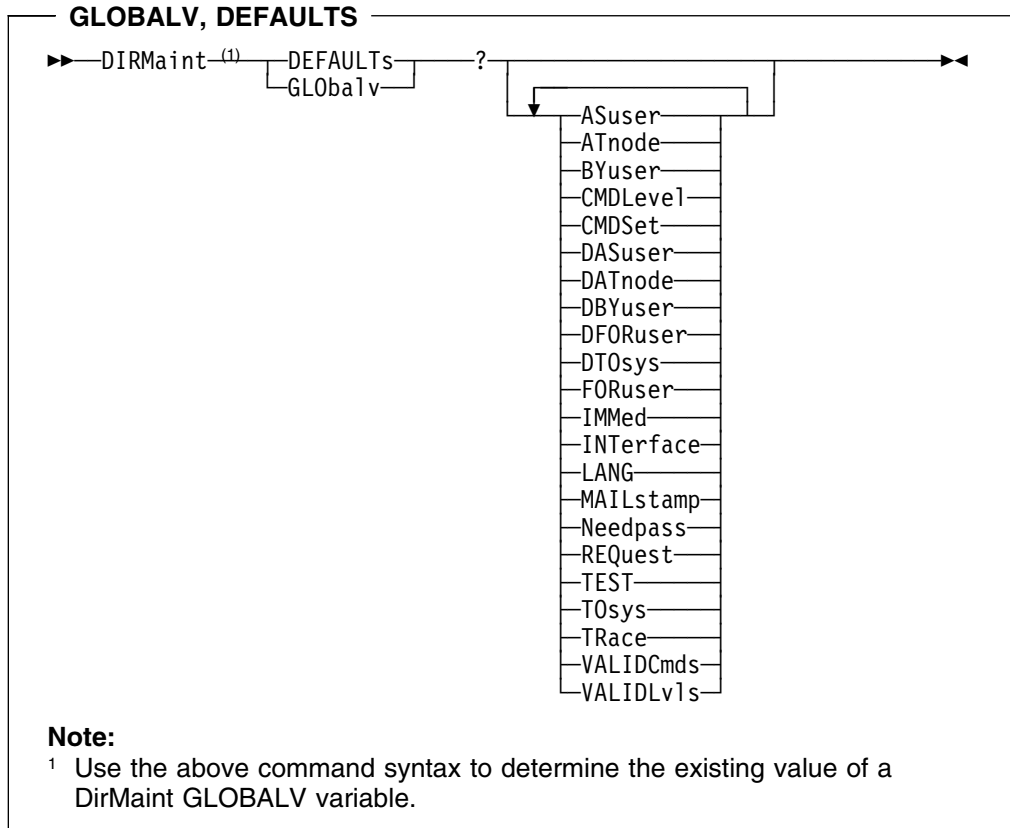
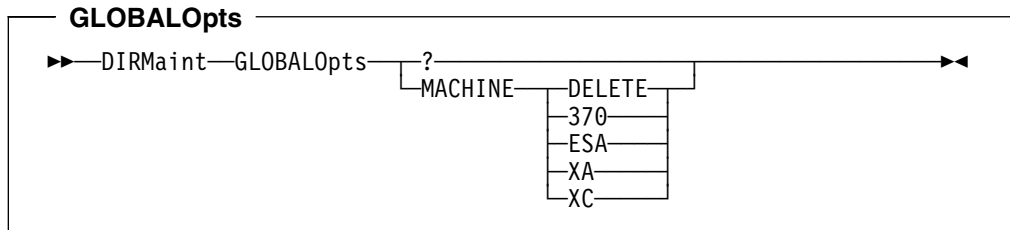
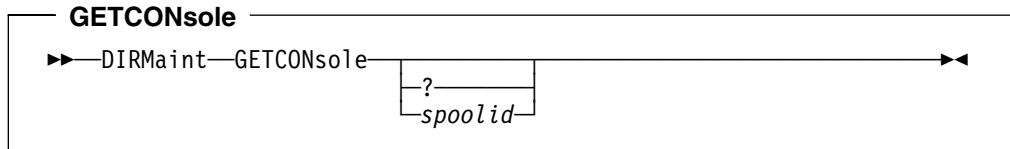
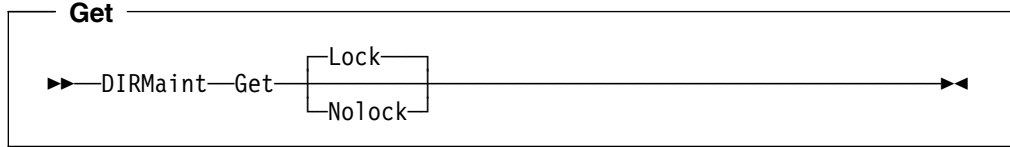
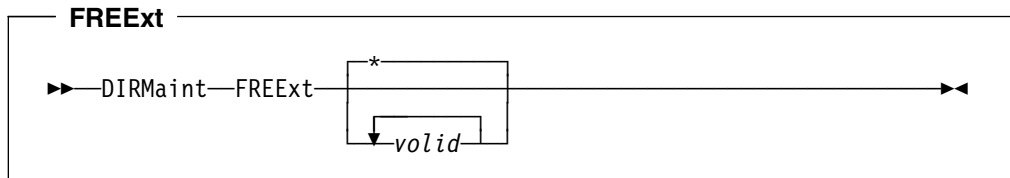


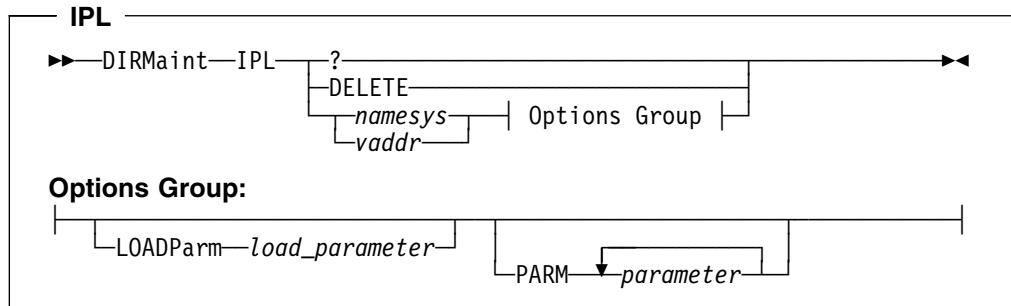
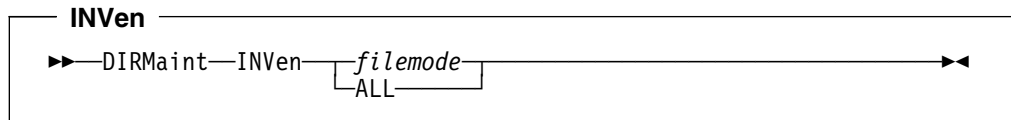
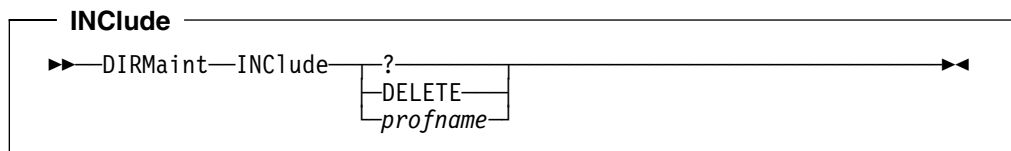
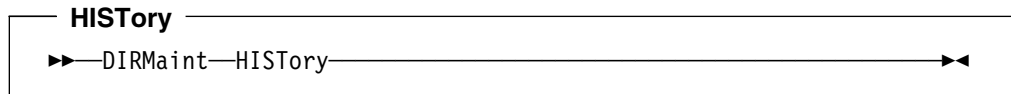
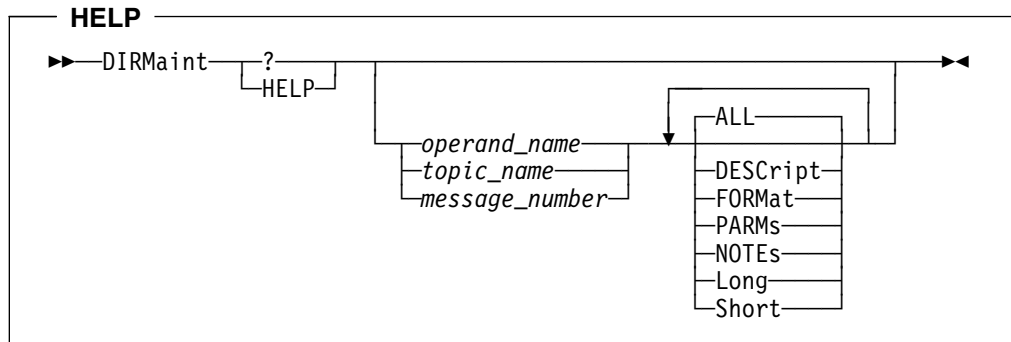
DIRECTORY

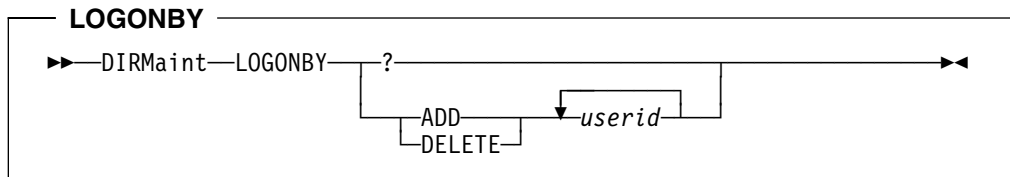
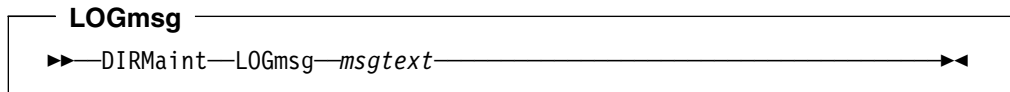
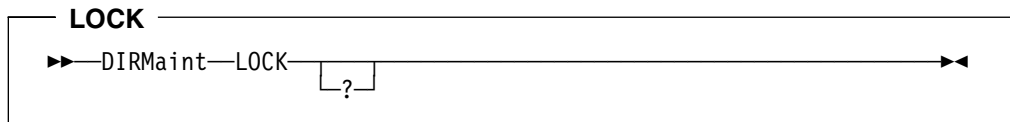
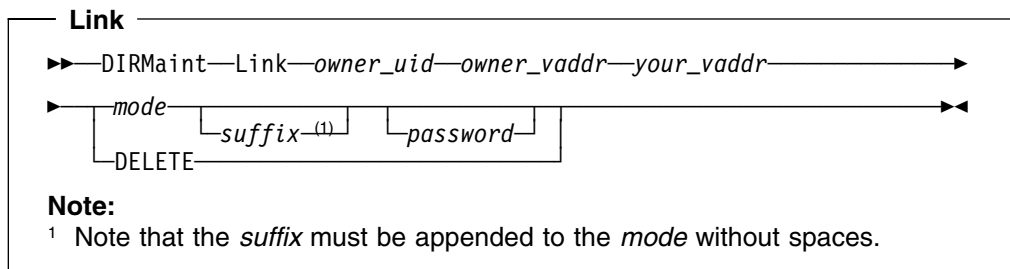
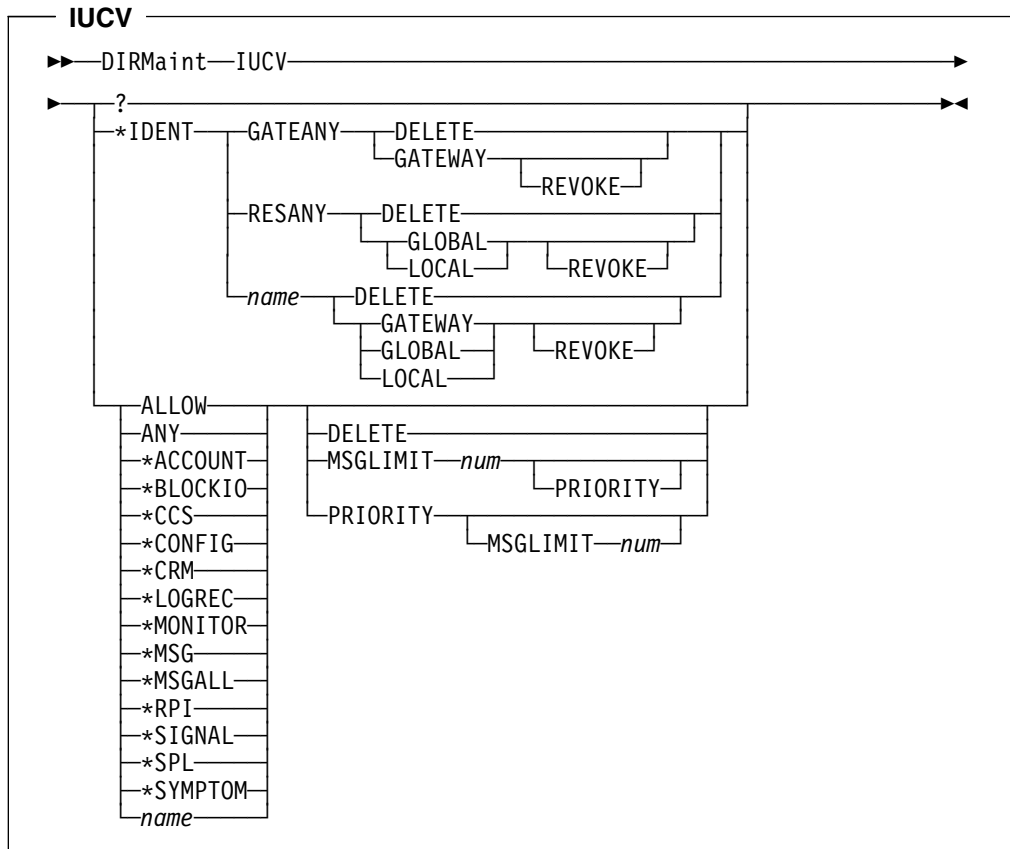


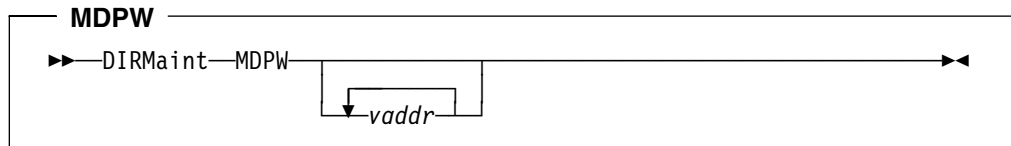
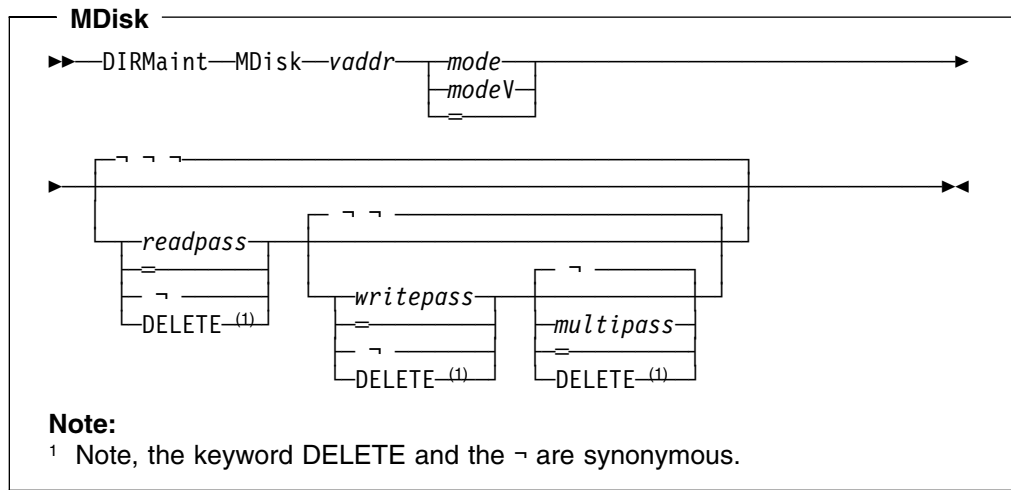
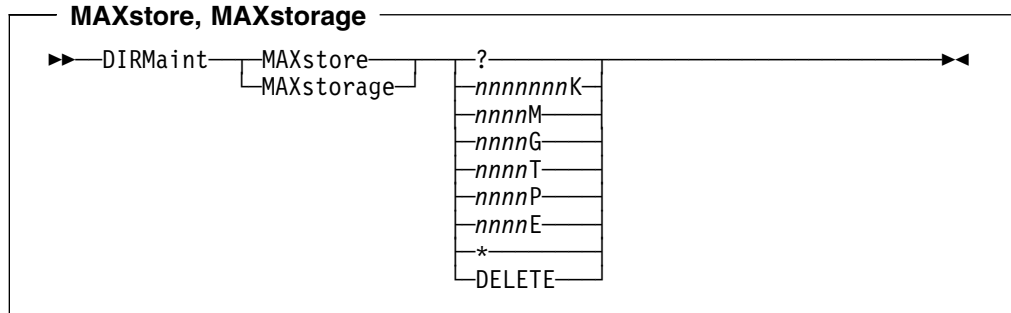
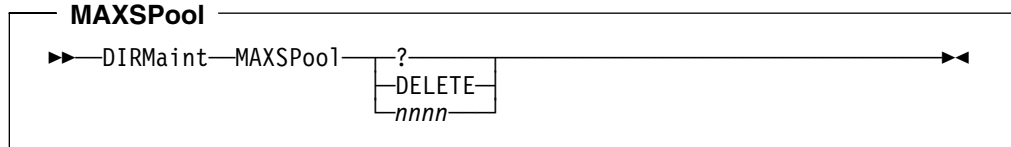
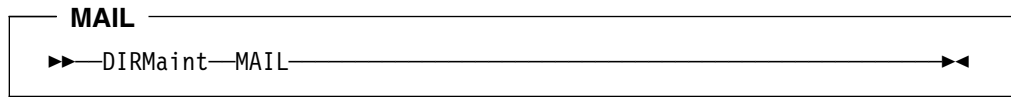
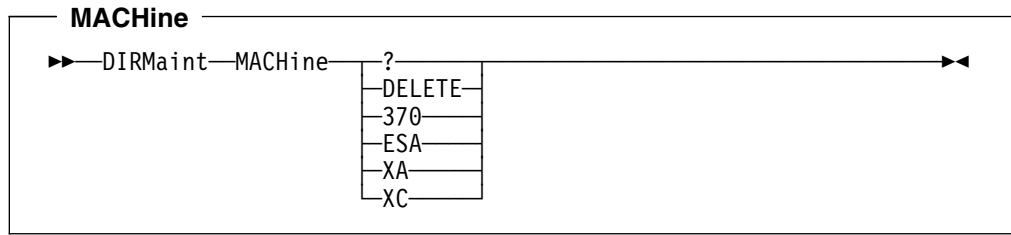


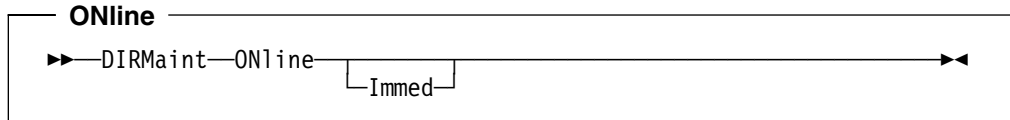
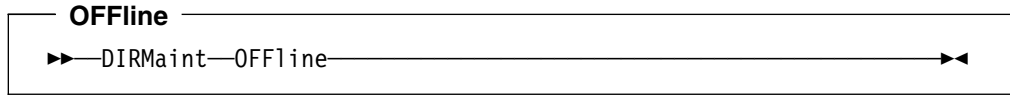
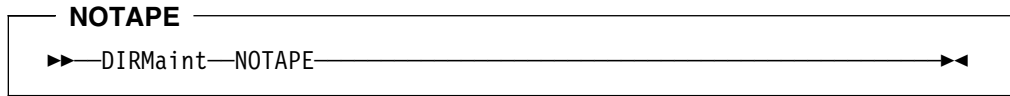
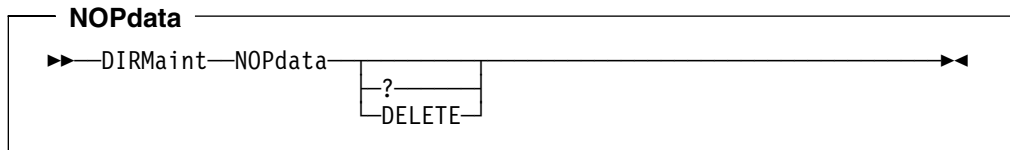
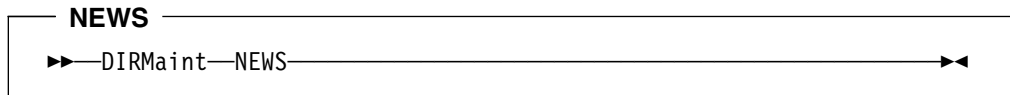
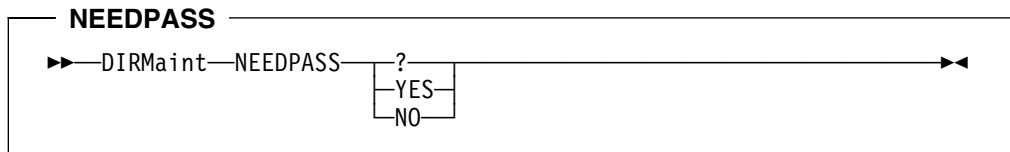
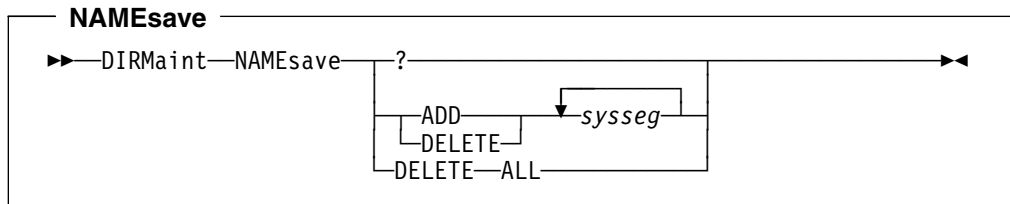
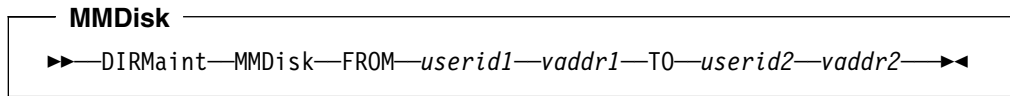
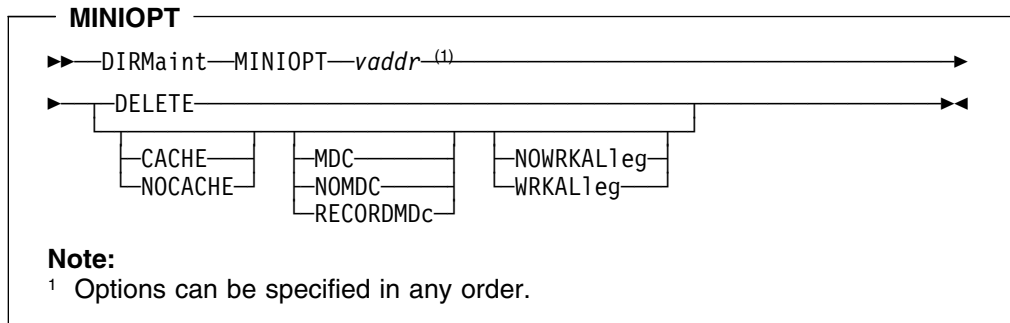


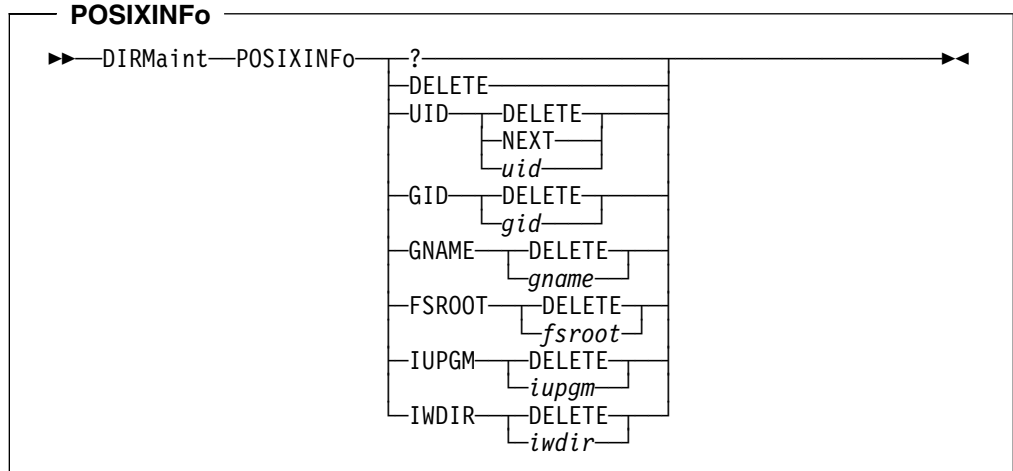
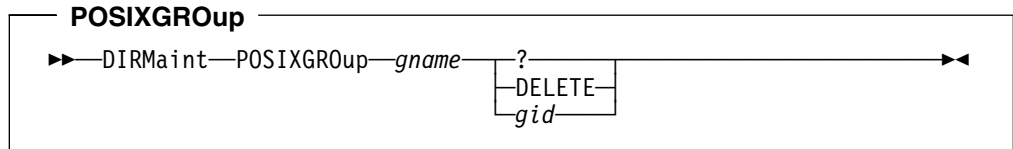
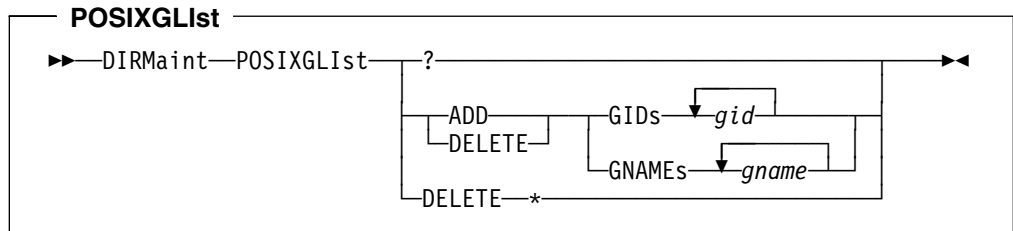
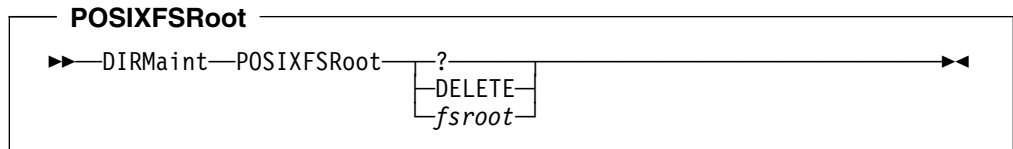
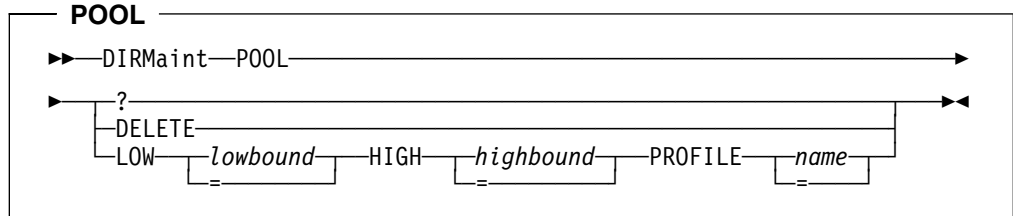
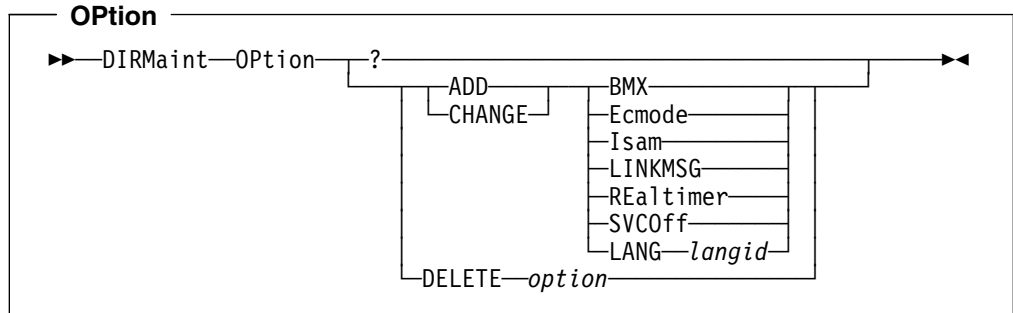


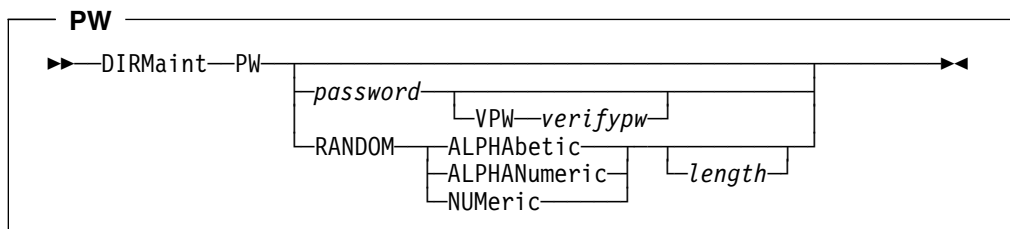
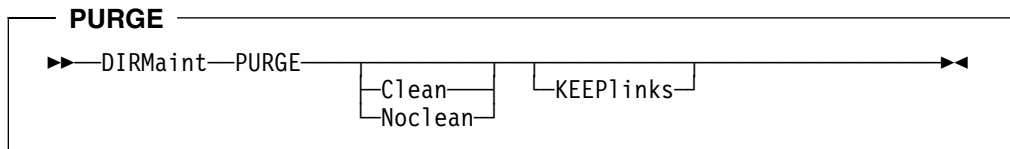
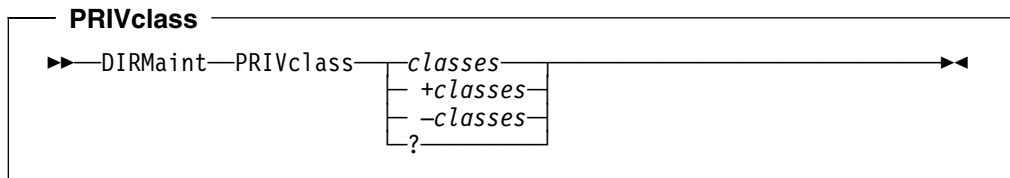
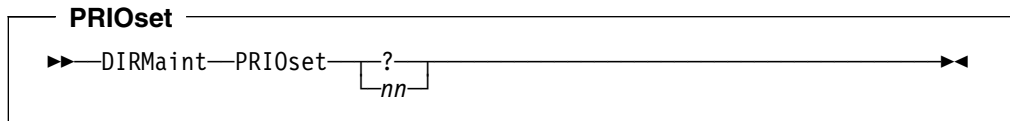
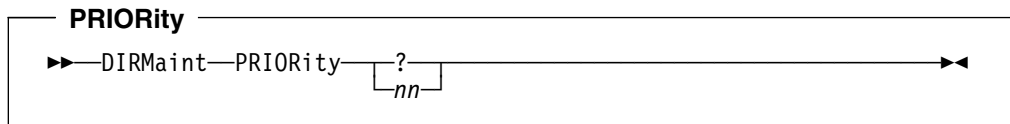
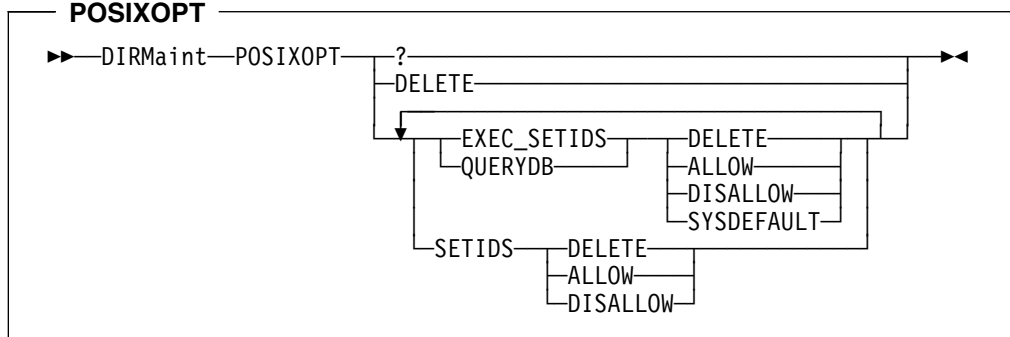
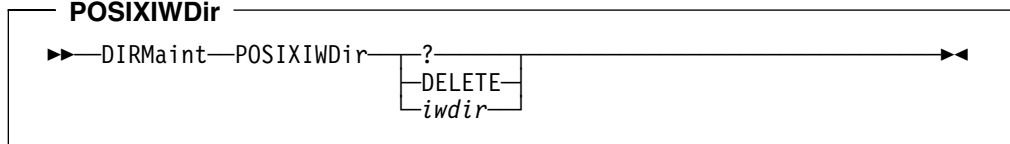
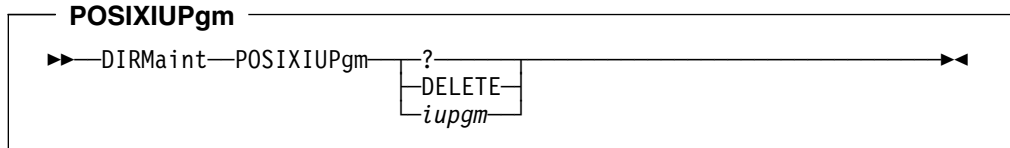




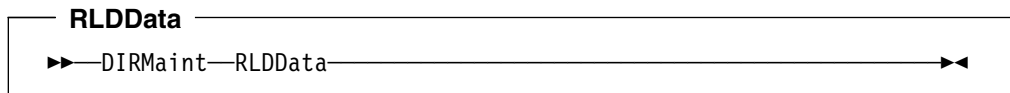
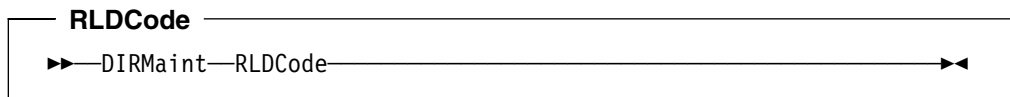
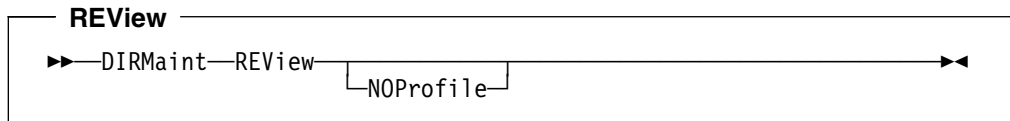
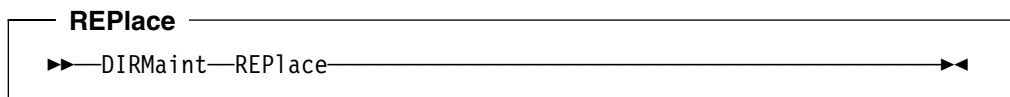
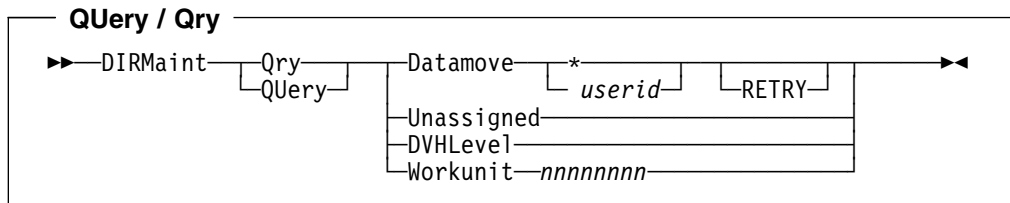
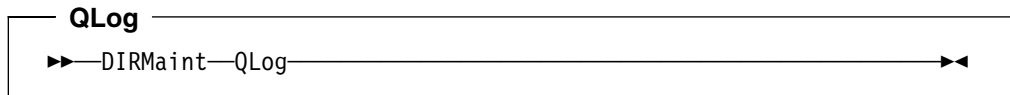
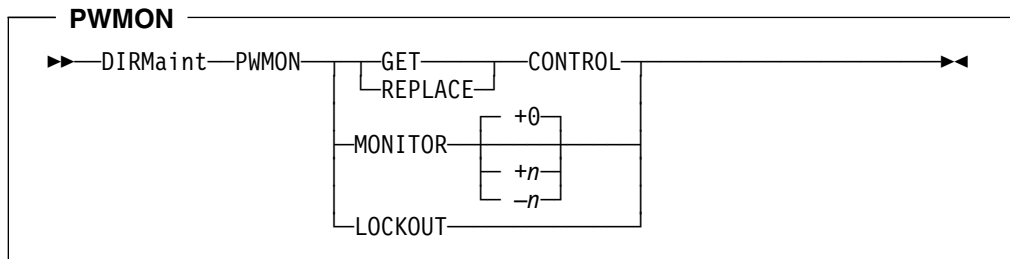
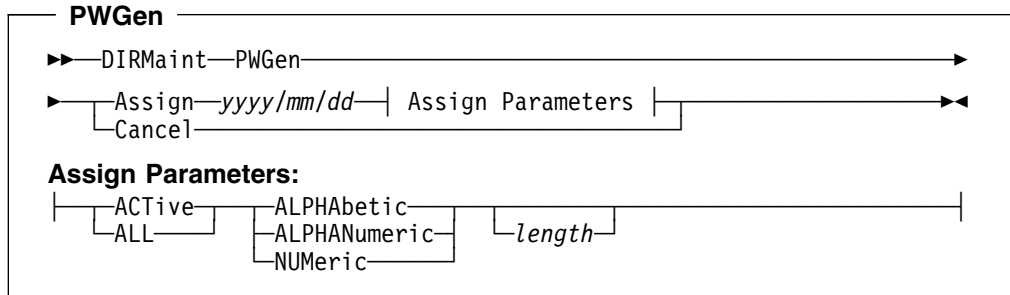
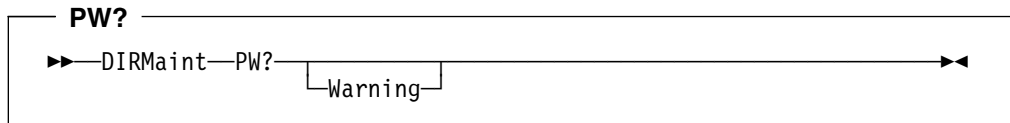


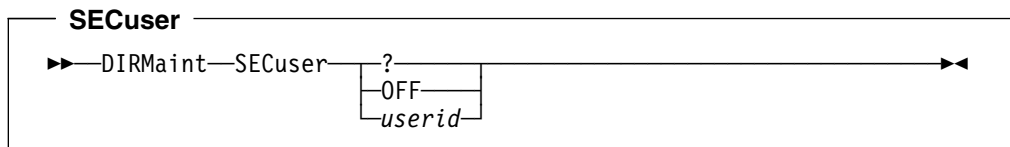
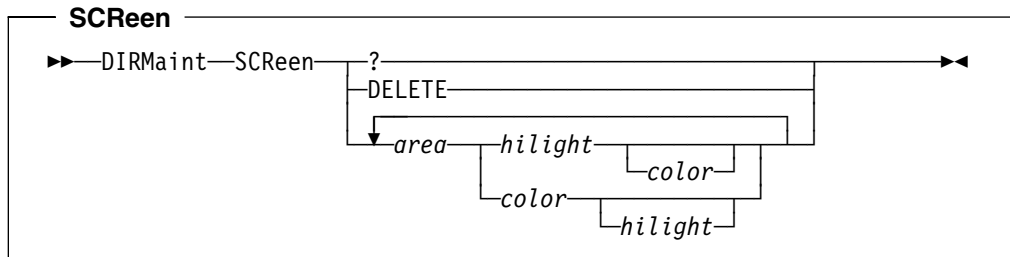
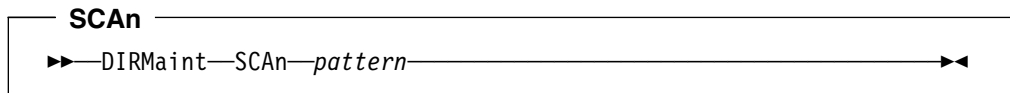
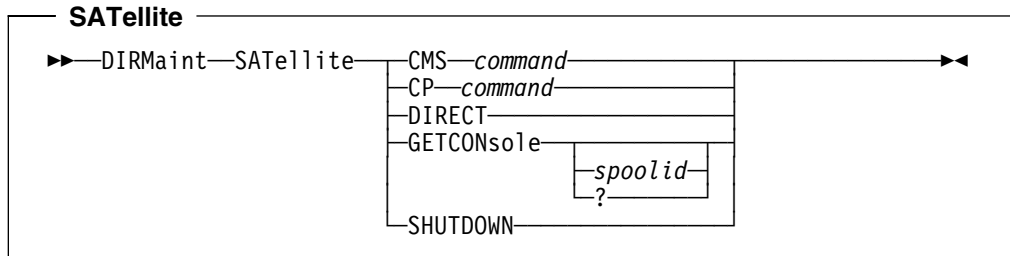
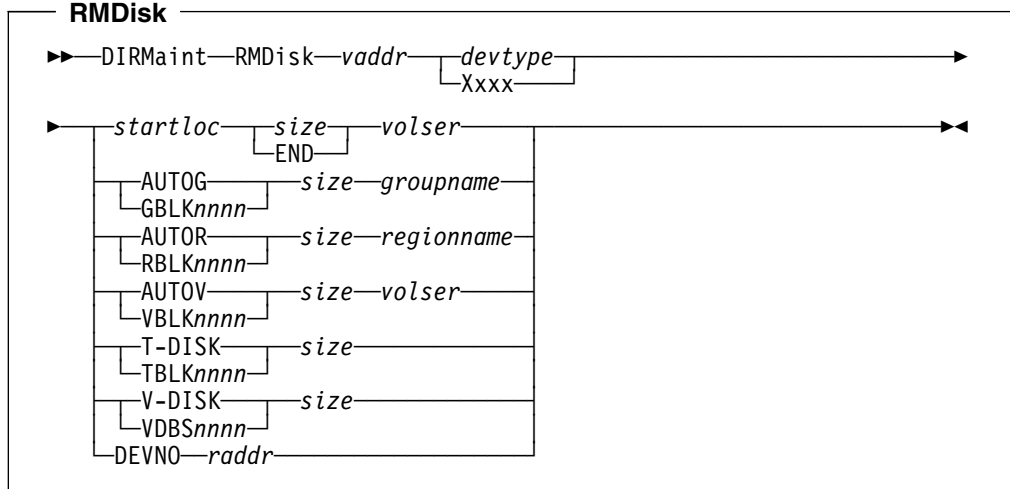
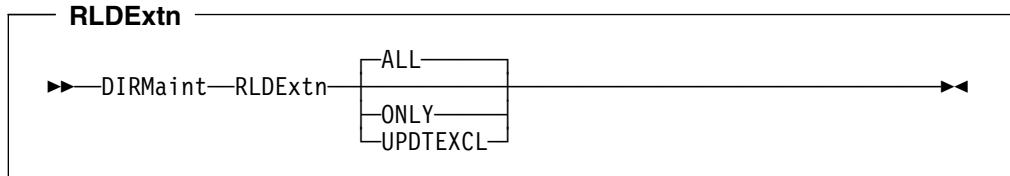


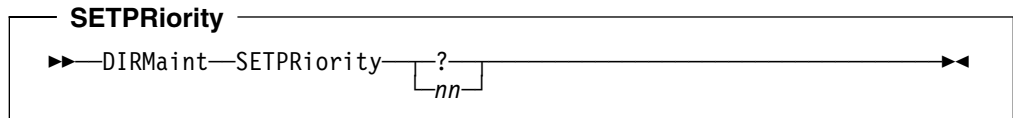
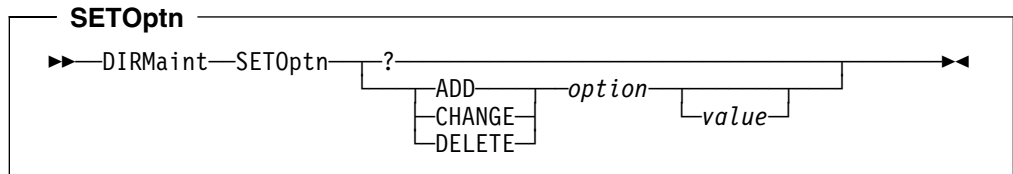
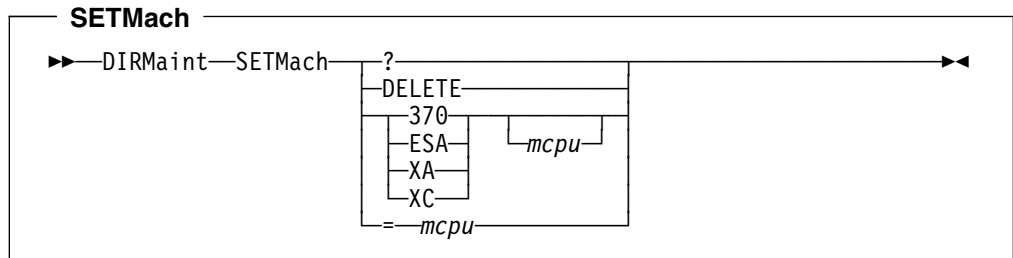
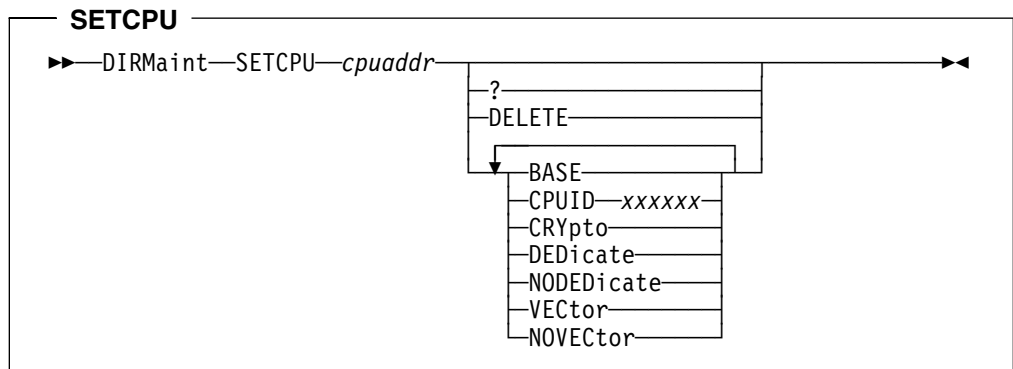
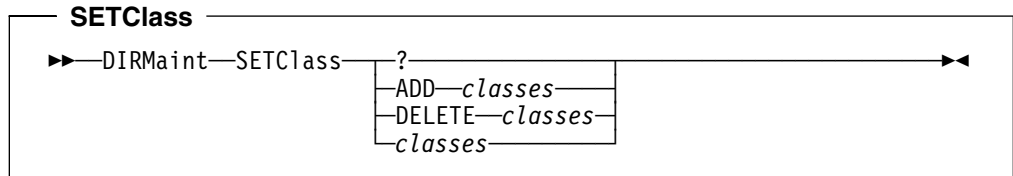
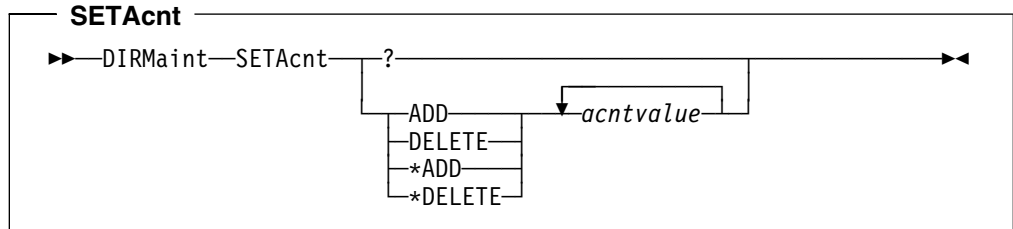
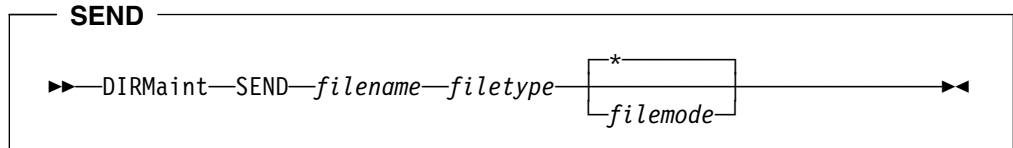


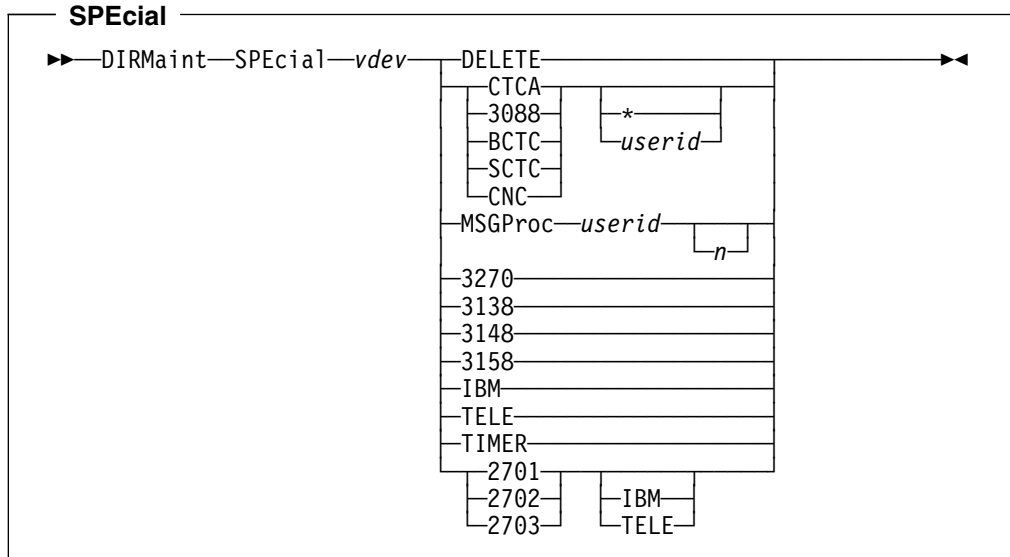
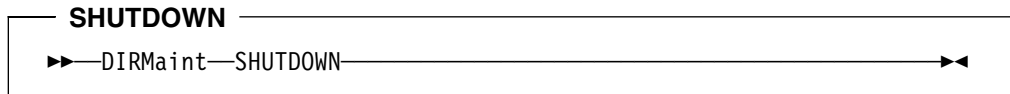
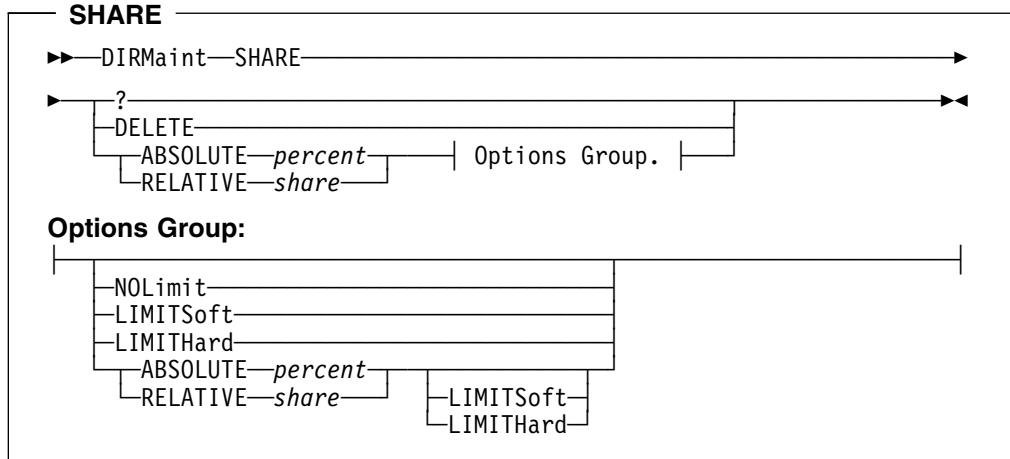
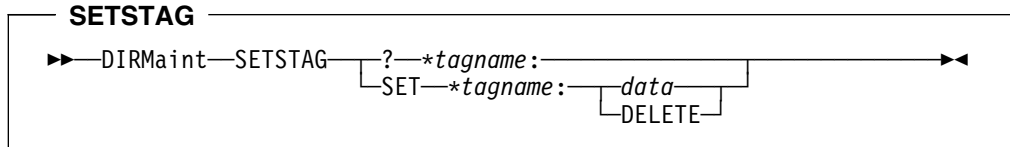
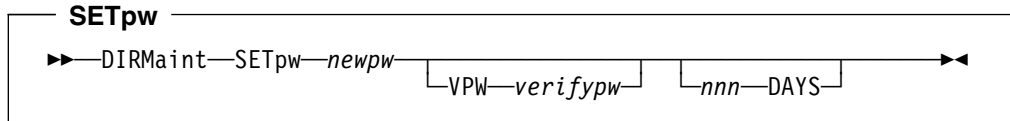


|

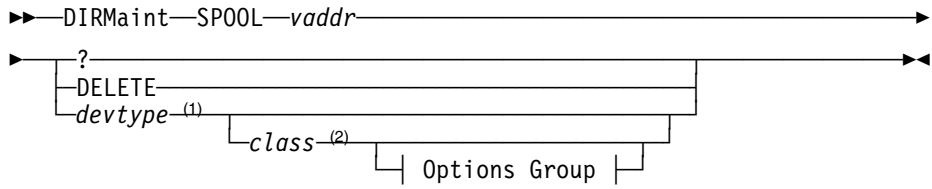




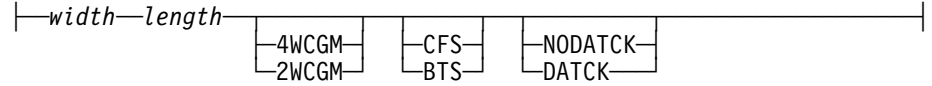




SPOOL



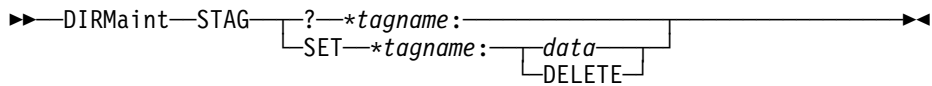
Options Group:



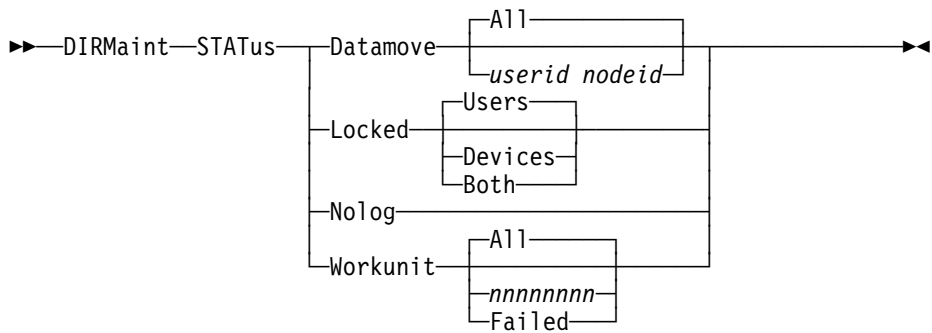
Notes:

- 1 Following a 2540 device one of these values must be specified indicating a reader or punch:
Reader or Punch
- 2 These variables are only recognized for a 3800 printer.

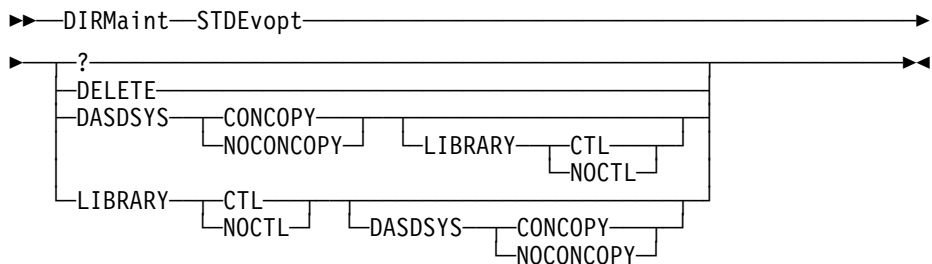
STAG

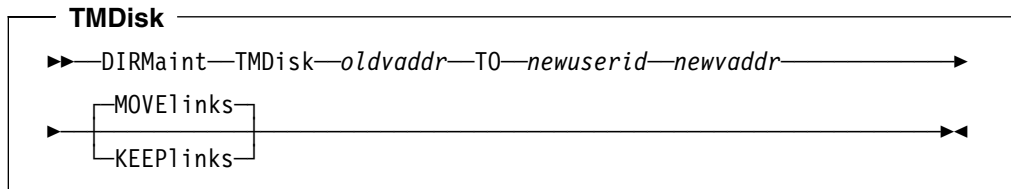
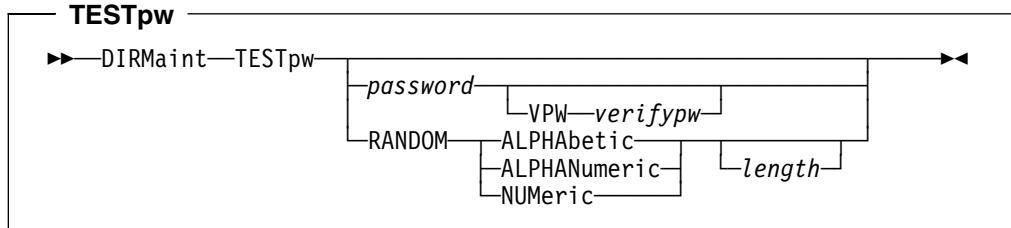
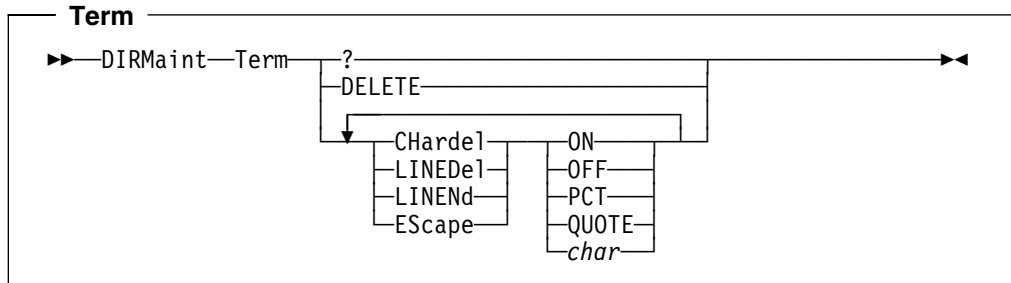
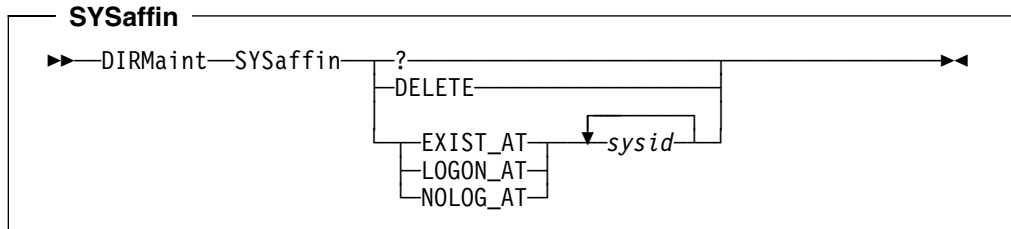
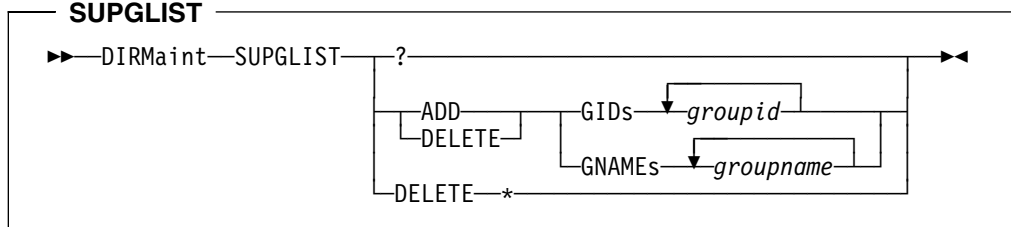
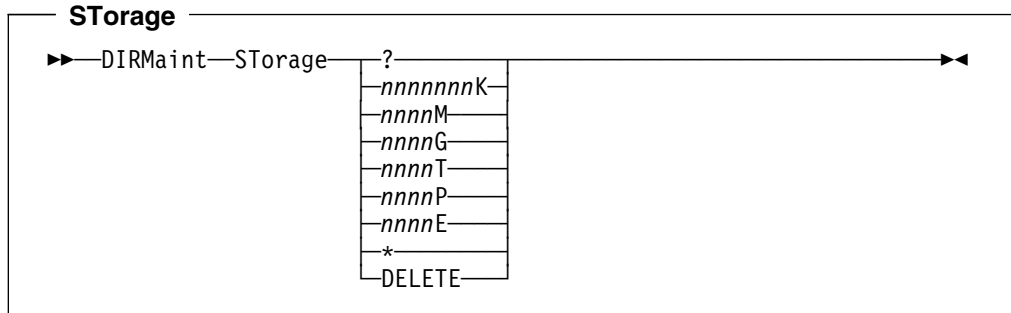


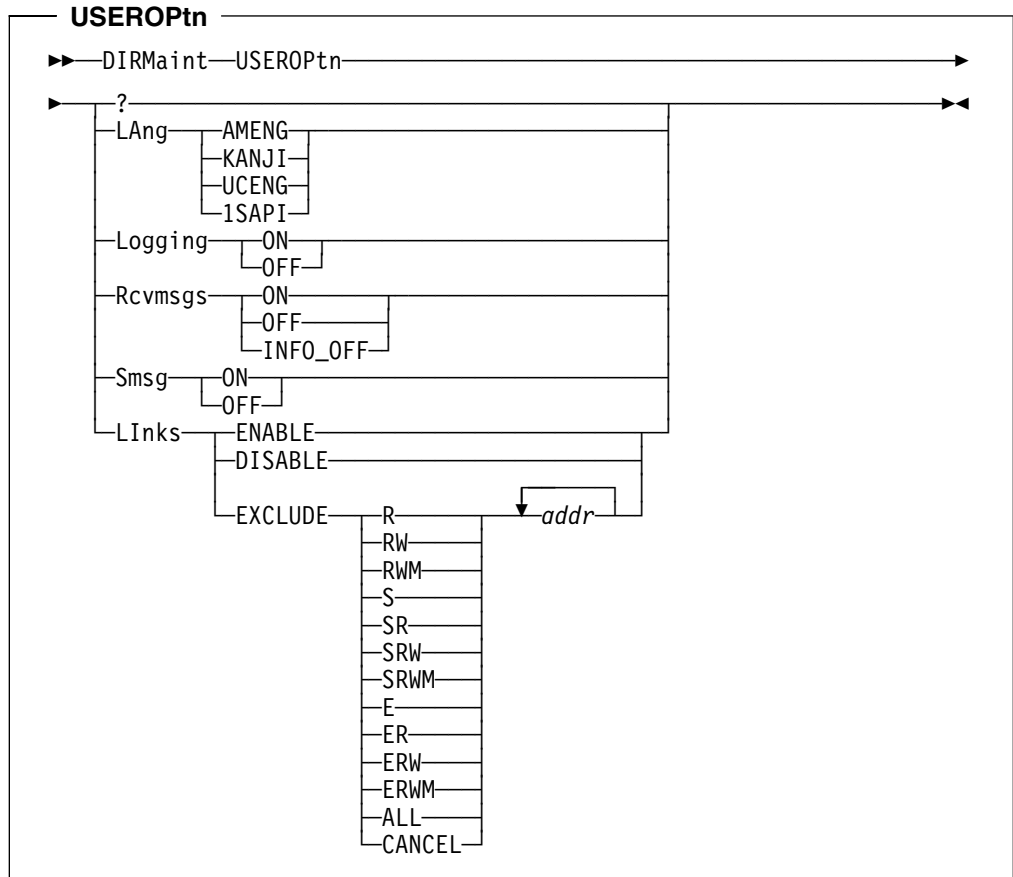
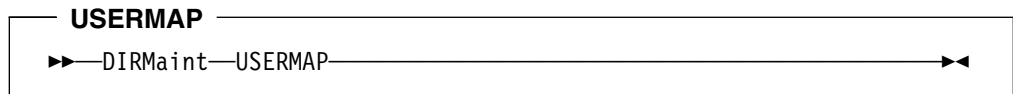
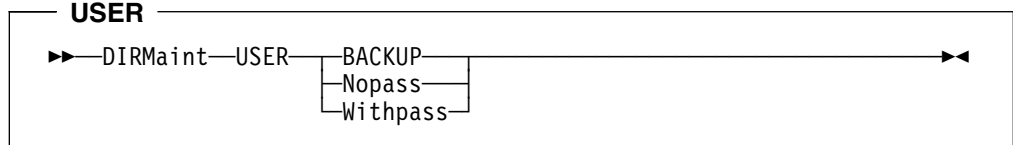
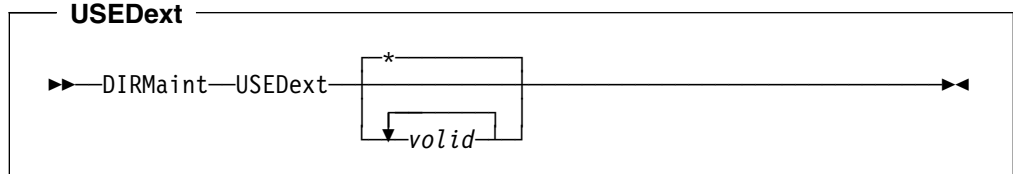
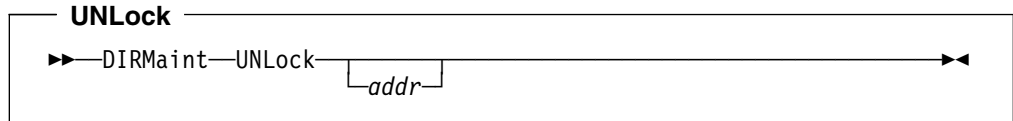
STATUS

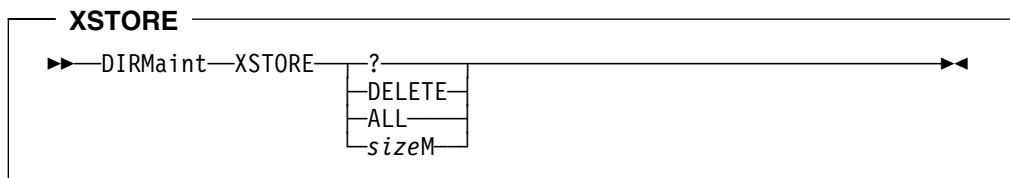
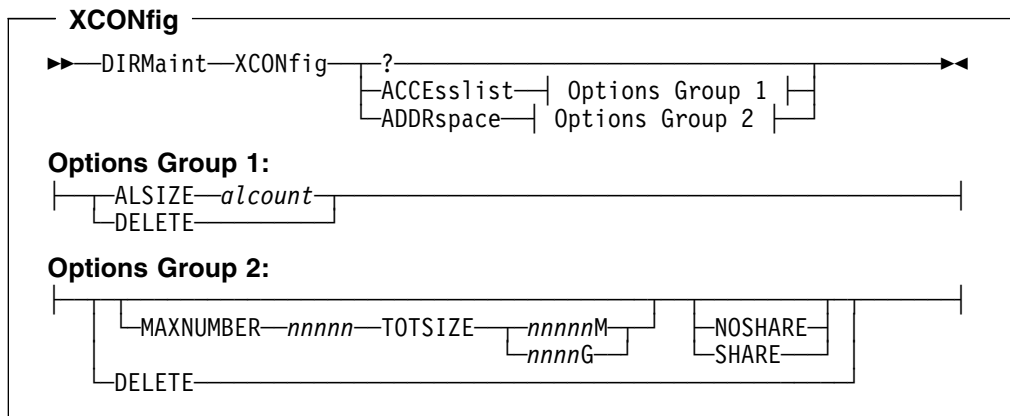
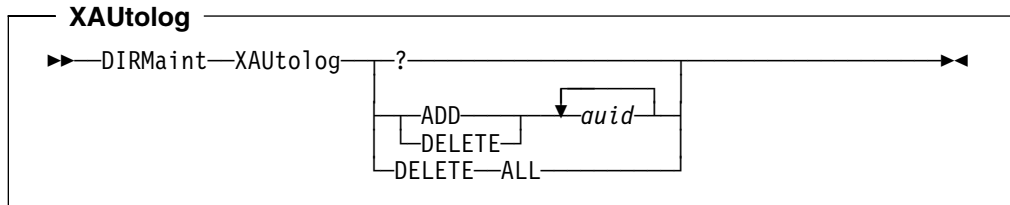
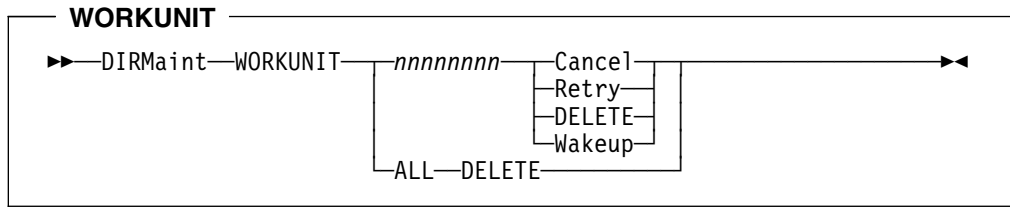


STDEvopt









Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes to the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300,
522 South Road
Poughkeepsie, NY 12601-5400
U.S.A.
Attention: Information Request

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities on non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information may contain sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to IBM's application programming interfaces. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Programming Interface Information

This book documents intended Programming Interfaces that allow the customer to write programs to obtain services of DirMaint.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, or other countries, or both:

- BookManager
- DFSMS/VM
- IBM
- IBMLink
- Library Reader
- Multiprise
- MVS
- MVS/SP
- Processor Resources/Systems Manager
- PR/SM
- RACF
- System/370
- System/390
- VM/ESA
- z/VM

Other company, product, and service names may be trademarks or service marks of others.

Glossary

A list of VM terms and their definitions is available through the online HELP Facility. For example, to display the definition of "cms," enter:

```
help glossary cms
```

You will enter the HELP Facility's online glossary file and the definition of "cms" will be displayed as the current line. When you are in the glossary file, you can also search for other terms.

If you are unfamiliar with the HELP Facility, you can enter:

```
help
```

to display the main HELP Menu, or enter:

```
help cms help
```

for information about the HELP command.

For more information about the HELP Facility, see the *z/VM: CMS User's Guide*. For more about the HELP command, see the *z/VM: CMS Command Reference*.

Bibliography

This bibliography lists the DirMaint books, and other books that you may find useful.

DirMaint Library

The following table lists the books in the DirMaint Release 5.0 library and their order numbers.

| Title | Order Number |
|--|---------------------|
| DirMaint: License Program Specifications | GC20-1837 |
| DirMaint: General Information | GC20-1836 |
| DirMaint: Tailoring and Administration | SC23-0533 |
| DirMaint: Command Reference | SC20-1839 |
| DirMaint: Messages | SC23-0437 |
| DirMaint: Diagnosis Reference | SC24-5883 |

z/VM Version 3 Release 1.0 Library

The following table lists books in the z/VM Version 3 Release 1.0 library that may be helpful.

| Title | Order Number |
|--|---------------------|
| z/VM: VMSES/E Introduction and Reference | GC24-5947 |
| z/VM: System Messages and Codes | GC24-5974 |
| z/VM: Planning and Administration | SC24-5948 |
| z/VM: CP Programming Services | SC24-5956 |
| z/VM: REXX/VM Reference | SC24-5963 |
| z/VM: CP Command and Utility Reference | SC24-5967 |
| z/VM: CMS Command Reference | SC24-5969 |
| z/VM: CMS Pipelines Reference | SC24-5971 |
| z/VM: XEDIT Command and Macro Reference | SC24-5973 |

VM/ESA Version 2 Release 4.0 Library

The following table lists books in the VM/ESA Version 2 Release 4.0 library that may be helpful.

| Title | Order Number |
|--|--------------|
| VM/ESA: VMSES/E Introduction and Reference | GC24-5837 |
| VM/ESA: Planning and Administration | SC24-5750 |
| VM/ESA: CP Command and Utility Reference | SC24-5773 |
| VM/ESA: CMS Command Reference | SC24-5776 |
| VM/ESA: REXX/VM Reference | SC24-5770 |
| VM/ESA: CMS Pipeline Reference | SC24-5778 |
| VM/ESA: XEDIT Command and Macro Reference | SC24-5780 |
| VM/ESA: CP Programming Services | SC24-5760 |
| VM/ESA: System Messages and Codes | GC24-5841 |

Other Related Books

The following table lists other books, outside the DirMaint and z/VM libraries, that may be helpful when using this book.

| Title | Order Number |
|--|--------------|
| VM/ESA: CMS Utilities Reference | SC24-5535 |
| RACF Security Administrator's Guide | SC28-1340 |
| RACF Macros and Interfaces | SC28-1345 |
| RACF Command Syntax Reference Booklet | SX22-0014 |
| C2/B1 Security Feature User's Guide for VM/ESA with RACF | SC24-5564 |
| C2/B1 Trusted Facility Manual for VM/ESA with RACF | SC24-5563 |
| DFSMS/VM Function Level 221 Storage Administration Guide and Reference | SH35-0111 |
| External Security Interface (RACROUTE) Macro Reference for MVS and VM | GC28-1366 |

CD-ROM

The following CD-ROM contains all the IBM libraries that are available in IBM BookManager format for current VM system products and current IBM licensed programs that run on VM. It also contains PDF versions of many z/VM publications and publications for some related IBM licensed programs.

- *Online Omnibus Edition: VM Collection, SK2T-2067*

Note: Only unlicensed publications are included.

Index

A

ACCOUNT command 16
ACIGROUP command 19
ACNTADD command 20
ACNTDEL command 22
ADD command 23
AMDISK command 27
APPCPASS command 34
AUTHBY command 37
AUTHFOR command 38
AUTHLINK command 39
AUTHSCIF command 41
AUTOLOG command 43

B

BACKUP command 45
BATCH command 47

C

CHECK command 49
CHKSUM command 51
CHNGID command 53
CHVADDR command 56
CLASS command 57
CLEANED command 58
CMDISK command 59
CMS command 66
CONSOLE command 67
COPIED command 68
CP command 69
CPU command 70
CRYPTO command 72

D

D8ONECMD command 103
DASD devices 283
DASDOPT command 74
DATAMOVE command 76
DATEFORM command 78
DEDICATE command 79
DEFAULTS command 81
DEFINESTAG command 85
DIRECT command 87
DIRECTORY command 88
DIREDDIT command 90
DIRM exec 8
DIRMAINT exec 8
DIRMAP command 91

DISTRIB command 94
DLINK command 95
DMDISK command 96
DROPBY command 98
DROPFOR command 99
DROPSCIF command 100
DSECUSER command 101
DUMP command 102

E

ELINK command 104
ENABLE command 106
EXECDROP command 107
EXECLOAD command 108
EXTNCHK command 109

F

FILE command 110
FREEEXT command 111

G

GET command 113
GETCONSOLE command 115
GLOBALOPTS command 116
GLOBALV command 117
glossary information 317

H

HELP command 121
HISTORY command 123

I

INCLUDE command 124
INVEN command 125
IPL command 126
IUCV command 128

L

LINK command 131
LOCK command 134
LOGMSG command 135
LOGONBY command 136

M

MACHINE command 137

MAIL command 138
MAXSPOOL command 139
MAXSTORAGE command 140
MAXSTORE command 140
MDAUDIT command 142
MDISK command 146
MDPW command 149
MINIOPT command 150
MMDISK command 152

N

NAMESAVE command 154
NEEDPASS command 156
NEWS command 157
NOPDATA command 158
NOTAPE command 159

O

OFFLINE command 160
ONLINE command 161
OPTION command 162

P

POOL command 168
POSIXFSROOT command 170
POSIXGLIST command 171
POSIXGROUP command 172
POSIXINFO command 173
POSIXIWDIR command 176
POSIXOPT command 177
POSIXUPGM command 175
PRIORITY command 179
PRIOSET command 180
PRIVCLASS command 181
PURGE command 183
PW command 185
PW? command 187
PWGEN command 189
PWMON command 191

Q

QLOG command 193
QRY command 194
QUERY command 194

R

REPLACE command 195
REVIEW command 197
RLDCODE command 198
RLDDATA command 199
RLDEXTN command 200

RMDISK command 202

S

SATELLITE command 206
SCAN command 207
SCREEN command 212
SECUSER command 214
SEND command 216
SETACNT command 217
SETCLASS command 219
SETCPU command 220
SETMACH command 222
SETOPTN command 224
SETPRIORITY command 230
SETPW command 231
SETSTAG command 232
SHARE command 234
SHUTDOWN command 236
SPECIAL command 237
SPOOL command 239
STAG command 241
STATUS command 243
STDEVOPT command 245
STORAGE command 246
SUPGLIST command 248
Synchronous Application Programming Interface (SAPI) 287
 DIRMSAPI EXECSTAMP 287
 rules 287
 sample programs 287
syntax diagrams, how to read the
 default 3, 4
 fragment 3, 5
 return arrow 3, 4
 symbols 3
 variable 3, 4
SYSAFFIN command 250

T

TERM command 252
TESTPW command 254
TMDISK command 256

U

UNLOCK command 258
USEDEXT command 259
USER command 261
USERMAP command 263
USEROPTN command 264

V

VERIFIED command 268

W

WORKUNIT command 269

X

XAUTOLOG command 271

XCONFIG command 273

XSTORE command 275

Communicating Your Comments to IBM

Directory Maintenance VM/ESA
Command Reference
Release 5.0
Publication No. SC20-1839-07

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:
1-607-752-2327 (US and Canada)
- If you prefer to send comments electronically, use this network ID:
 - pubrcf@vnet.ibm.com
 - USIB2L8Z@IBMAIL

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Readers' Comments — We'd Like to Hear from You

Directory Maintenance VM/ESA
Command Reference
Release 5.0
Publication No. SC20-1839-07

Overall, how satisfied are you with the information in this book?

| | Very Satisfied | Satisfied | Neutral | Dissatisfied | Very Dissatisfied |
|----------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| Overall satisfaction | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

How satisfied are you that the information in this book is:

| | Very Satisfied | Satisfied | Neutral | Dissatisfied | Very Dissatisfied |
|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| Accurate | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Complete | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Easy to find | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Easy to understand | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Well organized | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Applicable to your tasks | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? Yes No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.



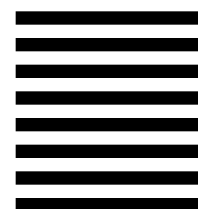
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Information Development
Department G60G
1701 North Street
Endicott, New York 13760-5553



Fold and Tape

Please do not staple

Fold and Tape



File Number: S370/S390-34
Program Number: 5748-XE4



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC20-1839-07





IBM Directory Maintenance VM/ESA

Command Reference

Release 5.0